Actes du Quatrième Colloque sur l'Optimisation et les Systèmes d'Information

COSI'2007

du 11 au 13 Juin 2007, Oran, Algérie

Organisation

Université des Sciences et de la Technologie d'Oran "Mohammed Boudiaf" (USTO-MB)

Président d'honneur

Professeur Mohammed BENSAFI, Recteur de l'USTO-MB, Oran, Algérie

Présidente du colloque

Nacéra BENAMRANE, Département Informatique, USTO-MB, Oran, Algérie

Membres

Hafida BELBACHIR
Fatima BENDELLA
Messabih BELHADRI
Abdelkader BENYETTOU
Mohamed BENYETTOU
Abdellah CHOUARFIA
Samira CHOURAQUI
Bachir DJEBBAR
Ahmed EC CHERIF
Hadria FIZAZI
Naima HADJ SAID
Karima KIES
Myriam NOUREDDINE
Fatiha SAÏS
Lynda ZAOUI

Comité de Pilotage

Farida ADMANE, CERIST, Alger, Algérie
Mohamed AIDENE, Université de Tizi-Ouzou, Algérie
Hafida BELBACHIR, Université des Sciences et Technologie d'Oran, Algérie
Mohand-Saïd HACID, Université de Lyon I, France
Lhouari NOURINE, Université de Clermont-Ferrand II, France
Jean Marc PETIT, INSA de Lyon, France
Bachir SADI, Université de Tizi-Ouzou, Algérie
Kamel TARI Université Bejaia, Algérie

Comité de programme

Président

Lakhdar SAÏS, CRIL - CNRS, Université d'Artois, France

Co-Président

Lhouari NOURINE, ISIMA, Université de Clermont-Ferrand II, France

Membres

AHMED NACER M. USTHB Alger (Algérie)
AHMED OUAMER R. Université de Tizi-Ouzou (Algérie)
AIDENE M. UMM Tizi-Ouzou (Algérie) AIDER M. USTHB Alger(Algérie)
AIT HADDADENE H. Université USTHB Alger (Algérie)
AMGHAR Y. INSA Lyon (France)
BAIOU M., Clermont-ferrand II (France)
BALINSKI M., Ecole Polytechnique de Paris (France)
BARA V., Clermont-ferrand II (France)
BELBACHIR H., USTO Oran (Algérie)
BENAMRANE N., USTO Oran (Algérie)
BENATALLAH B., University of New South Wales (Australie)
BENBERNOU S., Université de Lyon I (France)
BERRACHEDI A. USTHB Alger (Algérie)
BENYETTOU A. USTO Oran (Algérie)
BENYETTOU M. USTO Oran (Algérie)
BLIDIA M., Université de Blida (Algérie)
BOUCHEMAKH I., USTHB Alger (Algérie)
BOUZEGHOUB B., Universitérie) de Versailles (France)
CHOUARFIA A., USTO Oran (Algérie)
DE MARCHI F. Lyon I (France)
DE MONTGOLFIER F., Université Paris VII (France)
DJEBBAR B., USTO Oran (Algérie)
ENNAJI A. LITIS, Université de Rouen (France)
FRÉVILLE A. Université de Valenciennes (France)
HABIB M. Université Paris VII (France)
HACID M.S Université Lyon I (France)
HADDADI S. Université de Guelma (Algérie)
HAMADI Y., Microsoft Research Cambridge (Royaume Uni)
KHEDDOUCI H. Université Lyon I (France)
LETHI A.H., Metz (France)

MAHEY P. Université de Clermont-Ferrand II (France)
MESSABIH B. USTO(Algérie)
MESSINE F. ENSEEIHT Toulouse (France)
MONGEAU M. LAAS-CNRS, Toulouse (France)
PAUL C. Université Montpellier II (France)
PIECHOWIAK S., Université de Valenciennes (France)
PETIT JM., INSA Lyon (France)
RADJEF MS., Université de Univ-Béjaia (Algérie)
RAYNAUD O., Université de Clermont-Ferrand II (France)
SADI B. Université de Tizi-Ouzou (Algérie)
SCHNEIDER M. Université Clermont-Ferrand II (France)
SLIMANI Y. Université des Sciences de Tunis (Tunisie)
TALEB-AHMED A., Université de Valenciennes (France)
THIERRY E. ENS de Lyon (France)
TOUMANI F. Université de Clermont-Ferrand II (France)

Relecteurs additionnels

Gilles AUDEMARD, Alexandre AUSSEM, Sadok BEN YAHIA, Salem BENFERHAT, Belaid BENHAMOU, Lucas BORDEAUX, Fabrice BOUQUET, Nadjib BRAHIMI, Assef CHMEISS, Fabrice DERRIEN, Alain GELY, Gilles GONCALVES, Christophe GUOINAUD, Souhila KACI, Philippe LACOMME, Chu Min LI, Yannick LOISEAU, Bertrand MAZURE, Raoul MEDINA, Rokia MISSAOUI, Sylvie NORRE, Meltem OZTURK, Nathalie PERNELLE, Fatiha SAIS, Laurent SIMON, Charles SUQUET

Table des matières

Préface . . . . . page ix

Conférences invitées

– Qualité des données dans les systèmes d'information
Professeur Mokrane Bouzeghoub, PRiSM, Université de Versailles (France)

– Annotations par graphes étiquetés et recherche d'information par le contenu
Professeur Michel Chein, Lirmm, Université de Montpelier II (France)

– L'ordinateur peut-il faire des découvertes ?
Professeur Pierre Hansen, GERAD, HEC Montréal (Canada)

– Emerging innovations in multidimensional and multimodal biomedical imaging
Professeur Zaidi Habib, Division of Nuclear Medicine, Geneva University Hospital, Geneva, Switzerland

Articles

– The recent developments in batch-scheduling with graphs : new connections
Mourad Boudhar . . . . . page 1

– Application des techniques de fouille de données à l'analyse des fichiers log (Cas des serveurs proxy de l'UATL)
B. Ziani et Y. Ouinten . . . . . page 13

– An Overview On Predictive And Preemptive Maintenance Techniques In Manets
Sofiane Boukli Hacene and Ahmed Lehireche . . . . . page 25

– Nouvel algorithme de résolution d'un programme quadratique convexe à variables mixtes
S. Radjef et M.0. Bibi . . . . . page 33

– Optimisation d'un système d'inférence flou par la recherche taboue : Application à la navigation d'un robot mobile
F. Boufera, F. Debbat et F.Khelfi . . . . . page 45

– Vers une représentation formelle des schémas semi-structurés
Amar Zerdazi, Myriam Lamolle et Ludovic Menet . . . . . page 57

– Une approche basée sur les services web utilisant les spécifications ebXML pour e-commerce B2B
A. Raouf Kouhoul, Fethi Rabhi et Ryme Chelouah . . . . . page 71

– On b-colorings in regular graphs

Mostafa Blidia, Frédéric Maffray and Zoham Zemir ..... page 81

– Migrating Web Application towards Web Services Technologies
Mostefai Aek, Malki Mimoun and Bouchiha Djelloul ..... page 89

– Goal Programming and Scalarization Method for Solving Multiobojective Optimal Control Problem
M. Aidène et A. Merakeb ..... page 103

– GAE M2LL and RFOS load balancing algorithms on dynamic networks : an experimental study
J. M. Bahi, R. Couturier and A. Sider ..... page 115

– Méthode de la diagonale dominante pour un problème d'optimisation globale en dimension n sur un hyperrectangle
Le Thi Hoai An et Ouanes Mohand ..... page 129

– Etude de la transmission multicouche multicast avec différentiation de services
Hamdad Ismail, Elmagroud Rédouane et Mekkakia Maaza Zoulikha ..... page 139

– An Other Characterization of An Induced Subgraph of An Hypercube
Abdelhafid Berrachedi and Nawel Kahoul ..... page 151

– Analyse Discriminante Linéaire Bidimensionnelle Neuronale pour l'Identification
Chahrazed Rouabhia et Hicham Tebbikh ..... page 159

– Solving MAX-SAT Problems using a Scatter Search Metaheuristic
Dalila Boughaci , Belaïd Benhamou and Habiba Drias ..... page 167

– Caractérisation des graphes bγ parfaits
Mostafa Blidia, Noureddine Ikhlef-Eschouf and Frédéric Maffray ..... page 179

– Modélisation et Simulation de la Stérilisation Externe pour le Nouvel Hôpital d'ESTAING
Fateh Mebrek, Khaled Belkadi , Michel Gourgand et Alain Tanguy ..... page 191

– Towards semantic interoperability: An ontology mapping approach
Afaf Merazi, Sidi Mohamed Benslimane and Mimoun Malki ..... page 205

– Optimality and duality in nondifferentiable multiobojective programming under d_I-invexity
Hachem Slimani and Mohammed Said Radjef ..... page 217

– Optimisation Par Essaim de Particules : Application au Problème du Voyageur de Commerce
Fareh Abdelhak ..... page 231

– Modèle déformable appliqué à la détection du contour d'images médicales
M. Debakla, M. Benyettou, A. Bouadi ..... page 243

– Isometric eylvainbeddings of subdivided connected graphs
Meziane Aïder, S. Gravier and Kahina Meslem ..... page 257

– Résolution d'un système dynamique linéaire avec une commande polyèdrale
M. Aidène, B. Oukacha et F. Kara ..... page 265

– Indexation de vidéo couleur non compressée par le contenu à l'aide de la structure d'arbre R
Karim Abdelkader Henni et Lynda Zaoui ..... page 277

– La reconnaissance en-ligne des caractères isolés par les réseaux bayésiens dynamiques: segmentation en strokes élémentaires
Redouane Tlemsani et Abdelkader Benyettou ..... page 289

– Series Expansions for Optimization of Maintenance in Stochastic Reliability Models
Fazia Rahmoune and Djamil Aissani ..... page 303

– Gradual Uncertainty Association Rules Mining
Yassine Djouadi, Samir Redaoui, Karima Amroun ..... page 315

– Application de la méthodedes k-moyennes hiérarchique sur un corpus de textes arabes
Abdessalem Kelaiaia, Hayet Farida Merouani et Yamina Tlili-Guiassa ..... page 327

– Système de Classification Contextuel Appliqué à la Segmentation d'Images IRM
Abdelouahab Mousssaoui ..... page 337

– Optimisation Multi Critères par Algorithmes Génétiques du Design des Réseaux d'Osmose Inverse
Mehdi Metaiche et John Palmeri ..... page 349

– Méthode duale de support pour la résolution des problèmes quadratiques convexes à variables bornées
B. Brahmi et M. O. Bibi ..... page 365

– Intégration d'une Classe Virtuelle Synchrone dans une Plate Forme d'e-Learning
Louiza Belkhiri, Mohamed Ramdane et Rachid Ahmed-Ouamer ..... page 377

– Performances evaluation of the Labeled Neuro-Fuzzy Classifier
Mohamed Nemissi, Hamid Seridi and Mohamed Zine Aissaoui ..... page 389

– WordNet Based Multilingual Text Categorization
Bentaallah Mohamed Amine and Malki Mimoun . . . . . page 401

– Algorithmes de sélection dans un problème d'ordonnancement en présence des demandes stochastiques
Djamel Berkoune et Khaled Mesghouni . . . . . page 413

– Décodage acoustique phonétique de la langue arabe : Utilisation des réseaux gamma pour la reconnaissance des consonnes emphatiques
Abderrahmane Bendahmane, Abderrahmane Louni et Abdelkader Benyettou . . . . . page 421

– Application des algorithmes génétiques multi objectif au problème d'exploration des espaces de configurations des systèmes embarqués
A. Bengueddach, S. Niar et B. Beldjilali . . . . . page 433

– Vers une approche distribuée pour l'ordonnancement réactif sous incertitudes
S. Ourari, C. Briand et B. Bouzouia . . . . . page 449

– Un modèle de prédiction de la performance des processus basé sur les réseaux Bayésiens
M'hamed Haned, Sid-Ali Addouche et Abderrahman El Mhamedi . . . . . page 461

– Medical Image Segmentation using Enhanced Hoshen Kopelman algorithm
K. Kies and N. Benamrane . . . . . page 475

– Un méta-modèle pour l'étude de la cohérence globale des contraintes dans les bases de données
Djamel Berrabah, Faouzi Boufarès et Mohamed Badri . . . . . page 485

– Algorithme Evolutionnaire Quantique Pour la Recherche d'Ordre de Variables dans les Diagrammes de Décision Binaire
Abdesslem Layeb et Djamel-Eddine Saidouni . . . . . page 499

– A Cooperative feature selection approach based on island model
M. A. Esseghir, T. Hsu, G. Goncalves and Y. Slimani . . . . . page 509

– Réconciliation de références : une approche logique adaptée aux grands volumes de données
Fatiha Saïs , Nathalie Pernelle and Marie-Christine Rousset . . . . . page 521

– Learning based system for knowledge discovery from Nasopharyngeal cancer data
Khalid Benabdeslem , Mustapha Lebbah , Alexandre Aussem , Nadjim Chelghoum and Marilys Corbex page 533

– An incremental distributed algorithm for a partial Grundy coloring of graphs
Lyes Dekar, Brice Effantin and Hamamache Kheddouci . . . . . page 543

– Optimisation Multi-objectif pour l'Alignement Multiple de Séquences
N. Benlahrache et S. Meshoul . . . . . page 555

– Beetweeness centrality of metabolic network using Elementary Net Systems
Hafida Raaf, Saïd Abdeddaïm and Belhadri Messabih . . . . . page 567

– Performance analysis of feature interaction resolution methods
Ahmed Khoumsi . . . . . page 579

– Guided Local search for the Examination Timetabling Problem
Drifa Hadjidj and Habiba Drias . . . . . page 589

– Une approche infométrique pour l'analyse de données bibliographiques
Bilal Yalaou et Hakim Harik . . . . . page 601

– Towards easy prototyping of pattern mining problems
Frédéric Flouvat, Fabien De Marchi and Jean-Marc Petit . . . . . page 611

– A logarithm barrier method for semi-definite programming
Jean-Pierre Crouzeix and Bachir Merikhi . . . . . page 625

– Extending Greedy Local Search Algorithm to Unsatisfiability and Unvalidity
Gilles Audemard and Laurent Simon . . . . . page 647

Index des auteurs

Abdeddaïm Saïd 567
Abederrahman El Mhamedi 461
Addouche Sid-Ali 461
Afaf Merazi 205
Ahmed-Ouamer Rachid 377
Aidène Mohammed 103, 265
Aïder Méziane 257
Aissaoui Mohamed Zine 389
Amroun Karima 315
Audemard Gilles ..... **??**
Aussem Alexandre 533
Badri Mohamed 485
Bahi Jacques Mohcine 115
Beldjilali Bouziane 433
Belkadi Khaled 191
Belkhiri Louiza 377
Benabdeslem Khalid 533
Benamrane Nacera 475
Bendahmane Abderrahmane 421
Benguedach A. 433
Benhamou Belaïd 167
Benlahrache Nadira 555
Benslimane Sidi mohamed 205
Bentaallah Mohamed Amine 401
Benyettou Abdelkader 289, 421
Benyettou Mohammed 243
Berkoune Djamel 413
Berrabah Djamel 485
Berrachedi Abdelhafid 151
Bibi Mohand Ouamer 365, 33
Blidia Mostafa 81, 179
Bouadi Abed 243
Bouchiha Djelloul 89
Boudhar Mourad 1
Boufarès Faouzi 485
Boufera F. 45
Boughaci Dalila 167
Boukli Hacene Sofiane 25
Bouzouia B. 449
Brahimi B. 365
Briand Cyril ..... **??**
Chelghoum Nadjim 533
Chelouah Ryme 71
Corbex Marilys 533
Couturier Raphaël 115
Crouzeix Jean Pierre 625
De Marchi Fabien 611
Debakla M. 243
Debbat Fatima 45
Dekar Lyes 543
Djamil Aissani 303
Djouadi Yassine 315
Drias Habiba 167, 589
Effantin Brice 543
Elmagroud Rédouane 139
Esseghir M. A. 509
Fareh Abdelhak 231
Flouvat Frédéric 611
Goncalves Gilles 509
Gourgand Michel 191
Gravier Sylvain 257
Hadjidj Drifa 589
Hamdad Ismail 139
Haned M'hamed 461
Harik Hakim 601
Hayet Farida Merouani 327
Henni Abdelkader Karim 277
Hsu Tienté 509
Ikhlef-Eschouf Noureddine 179
Kahoul Nawel 151
Kara Fadhila 265
Kelaiaia Abdessalem 327
Kheddouci Hamamache 543
Khelfi Fayçel 45
Khoumsi Ahmed 579
Kies K. 475
Kouhoul Abderaouf 71
Lamolle Myriam 57
Layeb Abdesslem 499
Le Thi Hoai An 129
Lebbah Mustapha 533
Lehireche Ahmed 25
Louni Abderrahmane 421
Maffray Frédéric 81, 179
Malki Mimoun 89, 205, 401
Mebrek Fateh 191
Mekkakia Maaza Zoulikha 139
Menet Ludovic 57
Merakeb Abdelkader 103
Merikhi Bachir 625
Merouani Hayet Farida 327
Mesghouni Khaled 413
Meshoul S. 555
Meslem Kahina 257
Messabih Belhadri 567
Metaiche Mehdi 349
Mostefai Aek 89
Moussaoui Abdelouahab 337
Nemissi Mohamed 389
Niar S. 433

Ouanes Mohand 129
Oukacha Brahim 265
Ouinten Y. 13
Ourari S. 449
Palmeri John 349
Pernelle Nathalie 521
Petit Jean-Marc 611
Raaf Hafida 567
Rabhi Fethi 71
Radjef Mohammed Said 33, 217
Rahmoune Fazia 303
Mohamed Ramdane 377
Redaoui Samir 315
Rouabhia Chahrazed 159
Rousset Marie-Christine 521
Saidouni Djamel-Eddine 499
Saïs Fatiha 521
Seridi Hamid 389
Sider A. 115
Simon Laurent 647
Slimani Y. 509
Tanguy Alain 191
Tebbikh Hicham 159
Tlemsani Redouane 289
Yalaou Bilal 601
Yamina Tlili-Guiassa 327
Zaoui Lynda 277
Zemir Zoham 81
Zerdazi Amar 57
Ziani B. 13

Préface

Le Colloque sur l'Optimisation et les Systèmes d'Information (COSI) rentre cette année dans sa quatrième édition, après celles d'Alger (2006), de Bejaia (2005) et de Tizi-Ouzou (2004). Cette rencontre annuelle pluridisciplinaire est un modèle du genre, elle permet à des chercheurs Algériens et étrangers travaillant sur des thématiques transversales comme les systèmes d'information, l'optimisation combinatoire et la théorie des graphes de se rencontrer et de s'imprégner des dernières avancées technologiques. Les divers thèmes abordés admettent souvent des connexions à la fois théoriques - les problèmes posés sont souvent de nature combinatoire et difficiles - et pratiques - les applications cibles font souvent appel à diverses techniques issues des différents thèmes abordés durant le colloque.

COSI est un événement chaleureux et convivial favorisant les échanges scientifiques et humains ! Cette ambiance de travail fait que tout participant en garde un souvenir mémorable.

L'évolution du nombre de papiers soumis et du nombre de participants est une preuve indéniable de sa pérennité et de son succès. Trois cent deux intentions de soumission (résumés) et deux cent dix articles provenant de plus de quarante villes d'Algérie, d'une vingtaine de villes de France, de Tunisie, du Maroc, de Jordanie, d'Arabie Saoudite, du Canada, d'Australie et du Royaume-Uni ont été soumis cette année, soit une progression de 120 % environ. Cette évolution en terme de nombre soumissions s'est traduite (pour des raisons organisationnelles) par un taux d'acceptation de 26 % soit cinquante-cinq articles et vingt-cinq posters. Les articles couvrent les champs disciplinaires traditionnels de COSI mais aussi d'autres domaines connexes comme l'intelligence artificielle et le traitement d'images et de la parole.

Le programme du Colloque comporte également quatre conférenciers invités de renommée internationale. La première plénière, du Professeur Habib Zaidi de la division de médecine nucléaire de l'Hôpital Universitaire de Genève (Suisse), porte sur la problématique de l'imagerie médicale. Le Professeur Pierre Hansen du centre des hautes études commerciales de Montréal (Canada) répondra dans la seconde plénière à la question ń L'ordinateur peut-il faire des découvertes ? ż. La troisième, du Professeur Michel Chein de l'Université de Montpellier II (France), porte sur la recherche d'informations par le contenu. La quatrième plénière du Professeur Mokrane Bouzeghoub de l'Université de Versailles (France) concerne la qualité des données dans les systèmes d'information. Merci à eux de nous avoir fait l'honneur de leur présence à COSI.

Cette année, le colloque est précédé d'une école d'été de deux jours sur l'aide à la décision permettant ainsi aux jeunes chercheurs de suivre des cours avancés sur l'optimisation globale, la théorie des jeux, la fouille et les entrepôts de données, la satisfiabilité et la programmation par contraintes.

Nous tenons à remercier très chaleureusement l'équipe organisatrice et plus particulièrement la Présidente du colloque Nacéra Benamrane pour son dévouement et son travail remarquable pour la préparation de cette quatrième édition. Nous tenons à remercier tous les auteurs qui ont soumis des articles montrant ainsi la vitalité de cette rencontre scientifique. Nous tenons également à exprimer notre profonde reconnaissance aux membres du comité de programme et aux rapporteurs supplémentaires qui ont fait un excellent travail, et qui ont accepté la surcharge de travail induite par le nombre important de soumissions. Merci également à Pierre Colomb du LIMOS et à Olivier Roussel du CRIL pour leur aide précieuse dans la mise en forme des actes du colloque.

Enfin, n'oublions pas que l'organisation de ce colloque ne serait pas possible sans les aides de nos partenaires institutionnels et industriels. Qu'ils reçoivent ici notre profonde reconnaissance.

Tous nos voeux de réussite pour cette rencontre scientifique et longue vie à COSI !

Lhouari Nourine et Lakhdar Saïs

The recent developments in batch-scheduling with graphs : new connections

Mourad BOUDHAR

Faculté de Mathématiques, Université USTHB,
BP 32 Bab-Ezzouar, El-Alia 16111, Alger, Algérie
mboudhar@yahoo.fr

Abstract. In this paper, we aim at compiling the bibliography and the recent developments on scheduling problems involving batches and graphs. We consider the problem of minimizing the makespan on a batch processing machine in which jobs are not all compatible. Only compatible jobs can be included into the same batch. This relation of compatibility is represented by an undirected graph called "compatibility graph". The processing time of a batch is given by the processing time of the longest job in the batch. The capacity of the batch processing machine may be finite or infinite. We consider several models with varying batch capacities, processing times, release dates, or compatibility graphs. This document will be useful for possible research and development in this field.

Keywords batch processing machine, batch scheduling, job compatibilities, compatibility graph.

1 Introduction

Batch scheduling is a other extension of the classical scheduling models where at most one job can be treated at a time on any machine. It refers to the scheduling of groups of jobs that are processed together (material heated together in a furnace, material painted together, material transported together by an AGV, etc.). In these cases, the jobs in the same batch have to be compatible (similar physical or chemical properties, forms, colors, weights, etc.). The theory of batch scheduling is already well established. Some very special cases of job compatibility, the IF (Incompatible Families) structures and the CF (Compatible Families) structures have previously been treated.

The aspect of job compatibilities has long been neglected. In the last years it has been introduced a formal notion of a compatibility graph to the batch scheduling problem in various articles.

The job compatibilities are represented by an undirected graph (called compatibility graph) where the vertices are the jobs and two vertices are connected by an edge if the vertices (jobs) are compatible. Any batch has to contain only pairwise compatible jobs. Relating scheduling theory and graph theory appears to be an interesting and important new concept.

Other researchers [14, 16] consider a problem with graphs complementary to our graphs: edges indicate incompatibilities and they partition the jobs into stable sets. In this way, the clique partitioning problem is transformed to a coloring problem. For this chromatic scheduling there are usually no capacity constraints.

Five years ago there is no literature discussing batch scheduling problems of these types. Brucker et al. [12] have proved that the problem with any processing times, any release dates, $b = 2$ and where all jobs are compatible (graph G is complete) is NP-hard in the strong sense. In their paper, the authors do not introduce the notion of compatibility graphs. They have shown that the equivalent mirror image problem of minimizing the maximum lateness is NP-hard in the strong sense.

In this paper, we consider the problem of scheduling a set of jobs $J_1, \ldots, J_n$ (n is the number of jobs) on a single batch processing machine $B1$ in order to minimize the makespan C_{max} (final completion time). We suppose that there exists a relation of compatibility between jobs, where two jobs are said to be compatible if they can be processed simultaneously in the same batch. This relation is represented by a graph $G = (V, E)$ where V is the set of jobs and a pair of jobs is an element of the edge set E if and only if they are compatible. By definition, a batch forms a clique (a complete subgraph, not necessarily maximal) in the compatibility graph G. Since all jobs have to be executed, the problem is to find a decomposition of G into cliques $B_1, B_2, \ldots, B_l$ where l is not know in advance.

The capacity b of a batch (or machine), and hence the clique size, may be finite $b = k \geq 2$ (it can process up to k jobs simultaneously), variable, or infinite $b = \infty$ (it can process all jobs simultaneously). The job J_i has the processing time p_i and the processing time of a batch is equal to the maximum processing time of any job assigned to it. Also, release dates can be allowed for jobs. All jobs in a batch must be available at the same date, start at the same date and finish at the same date. Preemption is not allowed. For any schedule of jobs on batch B_j, $pB_j = \max_{J_i \in B_j} \{p_i\}$ is the processing time of batch B_j, $rB_j = \max_{J_i \in B_j} \{r_i\}$ is the release time of batch B_j, Cb_j the completion time of batch B_j and C_i the completion time of job J_i (which is equal to CB_j if $J_i \in B_j$).

Since all jobs have to be executed, the problem is to find a decomposition of G into cliques $B_1, B_2, \ldots, B_l$ where l is not know in advance. If release dates is not allowed, the schedule length C_{max} is $\sum_{j=1}^{l} pB_j$. These batches may then be scheduled in any order, without any idle time. Release dates change the nature of the problem substantially, as it is no longer sufficient to just partition the jobs into batches: we must also schedule these batches to take into account the release dates. The completion time of a feasible batch schedule must now satisfy the release date constraints $Cb_j \geq rB_j + pB_j$.

To specify a batch-scheduling problems with graphs, we have to describe a machine environment, job characteristics and an optimality criterion. We adopt the three field notation $\alpha/\beta/\gamma$ for classical scheduling problems, where α de-

scribes the machine environment (B for batch processing machine) and specifies the number of machines, β describes schedule and job characteristics (including graph compatibility and batch capacity), and γ describes the optimality criterion (example: $B1/G = (V, E), b, \ldots/C_{max}$).

It is the purpose of this article to provide the complete list of industrial applications (section 2) and of complexities for all possible capacities, processing times, release dates or compatibility graph: IF and CF structures (section 3), general graph (section 4), bipartite graph (section 5), split graph (section 6), complementary of bipartite graph (section 7), interval graph (section 8) and other structures of graphs (section 9).

2 Industrial Applications

The first application is describe in [27] and recover in [18]: A rolling-mill is describe where the metal goes through repeated cycles of rolling and heating to produce the final steed plates. In the heating phase, the metal in form of coils is piled up on a base and then heated together in a bell furnace. Each coil T_j has to be heated for at least p_i time units. A loading of the furnace represents a max-batch. The material heated together has to be compatible, which means in this case that the coils have to have similar heights. One may express these compatibilities by means of a tolerance height Δ and assign to a coil of height H the tolerance interval $[H - \Delta/2, H + \Delta/2]$. Two metal coils are compatible if their tolerance intervals intersect. For a given set of metal coils (jobs), one obtains an interval graph for the job compatibilities, and a clique in this graph defines an admissible batch for the furnace. the batch size is limited by a given finite capacity b. The objective is to minimize the total heating time C_{max}.

The second application has been studied in [11, 17] and also recover in [18]: A manufacturer is developing flow-lines for the production of metallic office equipment. In a first phase, a large number of holes of various shapes and sizes are to be punched into a metal sheet which is then bent in a second operation. the essential features of the hole-punching facility are displayed in Figure 1. Several heads are arranged in two parallel lines, each one equipped with a tool magazine, and the metal sheet is moving unidirectionally through the system. The heads can move perpendicular to the direction of the metal sheet and also have a small lateral movement of size Δ. Each hole-punching operation (of one or several holes) requires a stopover and positioning of the metal sheet in the system. During the repositioning of the metal sheet, each active head is moved to its working position. There is, therefore, no additional tool changing time. During a stopover, all the holes that can be reached by the various heads (equipped with the appropriate tool) are grouped together in a work phase (forming a batch) and punched simultaneously. The batches are again max-batches since the longest hole-punching operation defines the duration of the work phase.

In this application the job compatibilities are defined by the geometry of the system. We may define the compatibility graph $G = (V, E)$ as follows. A node of V is a pair (h, H), where h is a hole and H is a head whose tool magazine

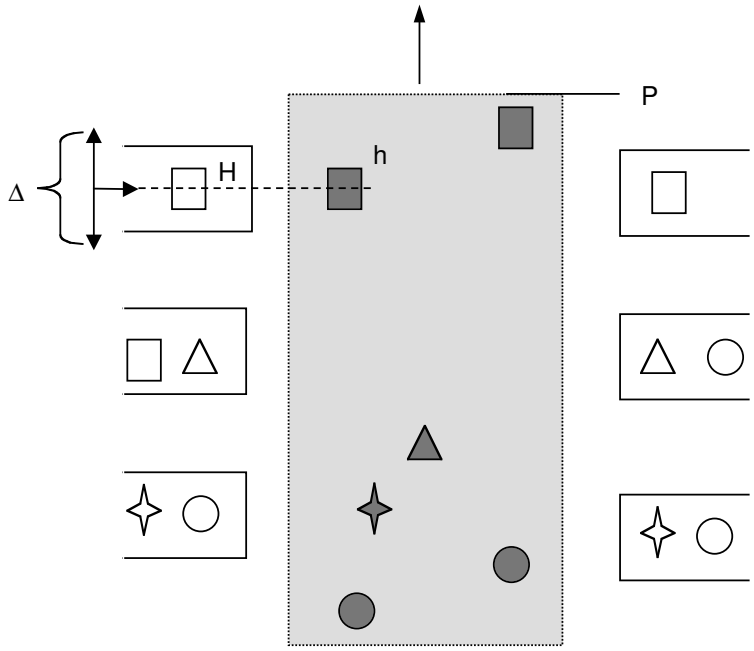


Fig. 1. A manufacturing system with 6 hole punching heads

contains the suitable tool for h and is situated on the appropriate side of the two-line system. It is assumed that for each hole h, exactly one head H has been pre-assigned. With each node (h, H) we associate the topmost position P of the metal sheet for which h is in front of the central position of H. Then hole h can be made by head H for all positions of the metal sheet in interval $I(h, H) = [P - \Delta/2, P + \Delta/2]$. If two such intervals $I(h, H)$ and (h', H') intersect, then the two corresponding holes $h \neq h'$ can be punched in one work phase, provide that $H \neq H'$. Because of these restrictions ($H \neq H'$), the compatibility graph is only 'almost' an interval graph. The objective is to minimize the total hole-punching time C_{max}.

Another application is also describe in [11, 17]: batches correspond to a separate working phase in an industrial process. A concrete application of a graph has been described in the articles above. A metal sheet passes through a k-head machine for a hole punching procedure. Depending on the position of the holes on the sheet, batches of holes are punched simultaneously in each working phase. If holes are represented by vertices of a graph G and its edges indicate compatibilities between holes, the resulting graph G is a k-partite graph (bipartite graph for $k = 2$ and can also be complete). The processing time of punching is, obvi-

ously, the same of all holes, but can be different if we consider the case where some holes are obtained in several punching procedures.

3 *IF* and *CF* structures

In this case the jobs are divided in families: Incompatible Families (IF) or Compatible Families (CF). The processing times of the jobs in the same family are identical and the jobs in the same family are all compatible. In all papers met in literature the IF and CF structures cannot be regarded as structures of graphs since the graph properties are not used. The three below tables summarize the main results.

Table 1. Polynomial problems for the IF and CF structures

Type	Algorithmic complexity	Reference
$B1/IF, b/\Sigma w_i C_i$	$O(nlogn)$	[30]
$B1/IF, b/C_{max}$	$O(n)$	[30]
$B1/IF, b/L_{max}$	$O(nlogn)$	[30]
$B1/IF, b, r_i/C_{max}$	$O(nlogn)$	[30]
$B1/CF, b = 2/\Sigma C_i$	$O(f^2)$ (f is the number of families)	[20]
$B1 \longrightarrow P1/IF, b/C_{max}$	$O(nlogn)$	[1]

$B1 \longrightarrow P1$ indicates that we are a flow shop problem with a batch processing machine in the first stage and a ordinary machine in the second stage.

Table 2. NP-hard problems for the IF and CF structures

Type	Proposed algorithms	Reference
$B1/IF, b/\Sigma T_i$	Dynamic Programming Heuristic	[26]
$Bm/IF, b/\Sigma w_i C_i$	Heuristic	[30]
$Bm/IF, b/C_{max}$	Heuristic	[30]
$Bm/IF, b/L_{max}$	Heuristic	[30]
$B1/IF, b, r_i/L_{max}$	Heuristic	[30]
$B1/IF, b/\Sigma \lvert C_i - D \rvert$	Dynamic Programming	[22]

Of course, each algorithm proposed is specific for the considered problem.

4 General graphs

Batch scheduling with a general compatibility graph G has been analyzed for the first time in [10]. It has been reported in this paper that the problem with

Table 3. Open Problems for the IF and CF structures

Type	Proposed Algorithms	Reference
$B1/CF, b/\Sigma C_i$	Dynamic Programming	[12]
$B1/CF, b/\Sigma\|C_i - D\|$	Dynamic Programming	[22]
$B1/CF, b/\Sigma C_i$	Dynamic Programming	[13]
$B1/CF, b/\Sigma C_i$	Optimal + Heuristic	[20]

an arbitrary compatibility graph G and capacity 2, denoted $B1/G = (V, E), b = 2/C_{max}$ can be solved in polynomial time of order $O(n^3)$ as a maximum weight matching problem. Note that the sub-problem with an identical processing time denoted $B1/G = (V, E), b = 2, p_i = 1/C_{max}$ can be solved in polynomial time of order $O(n^{2.5})$ as a maximum cardinality matching problem. In this paper, have been established new complexity results for the problem of minimizing the makespan on a single batch processing machine, in which jobs are not all compatible. Tables 4 and 5 summarize the problem types and their complexity status.

Table 4. Polynomial problems for general graphs

Problem	Complexity
$B1/G = (V, E), b = 2/C_{max}$	$O(n^3)$
$B1/G = (V, E), b = 2, p_i = 1/C_{max}$	$O(n^{2.5})$

Indeed, an optimum solution can be obtained by transforming the problem into a maximum weight matching problem as follows. Assign the weights $min\{p_i, p_j\} = p_i + p_j - max\{p_i, p_j\}$ to the edges $(i, j) \in E$ and solve the maximum weight matching problem in G. Then for each edge of the matching, process the corresponding two jobs in the same batch. The other jobs are processed as single job batches.

Table 5. NP-hard problems for general graphs

Problem
$B1/G = (V, E), b = k, p_i = 1/C_{max}$ $(k \geq 3)$
$B1/G = (V, E), b = 2, r_i, p_i = 1/C_{max}$
$B1/G = (V, E), b \geq n, p_i = 1/C_{max}$
$B1/G(K_b^1, \ldots, K_b^s), b, r_i, p_i = 1/C_{max}$

$G(K_b^1, \ldots, K_b^s)$ indicates a graph having a partition into s cliques with b vertices.

For the NP-hard problems, severals algorithms, exact of type branch and bound and heuristics, are presented in the above paper.

5 Bipartite graphs

In this section we can restrict our attention to special graphs. We shall concentrate on bipartite graphs $G = (S_1 \cup S_2; E)$ as compatibility graphs. The jobs S_1, respectively S_2, of G form a stable, hence are pairwise incompatible. Therefore, each job of S_1, respectively S_2, belongs to a separate batch, but may eventually be combined with some jobs of the clique S_2, respectively S_1. Since the maximum size of the batches is equal to 2, all constraint capacities of the batch processing machine are equivalent to $b = 2$.

The special cases with bipartite compatibility graph are considered in [5].

Table 6. Summary of results for bipartite graphs

Problem	Complexity
$B1/G = (S_1 \cup S_2, E), b/C_{max}$	$O(n^{2.5})$
$B1/G = (S_1 \cup S_2; E), b = 2, r_i, p_i = 1/C_{max}$	NP-hard in the strong sense
$B1/G = (S_1 \cup S_2; E), b = 2, r_{S_1}, r_{S_2} = 0, p_i = 1/C_{max}$	$\leq O(n^{2.5} logn)$
$B1/G = (S_1 \cup S_2; S_1 \times S_2), b = 2, r_i, p_i = 1/C_{max}$	$O(nlogn)$

$r_{S_1}, r_{S_2} = 0$ indicates that the release times for the jobs of the stable S_1 are arbitrary and the release times for the jobs of the stable S_2 are all equal to 0. $B1/G = (S_1 \cup S_2; S_1 \times S_2)$ indicates a bipartite complete graph.

It have also presented for the general problem with equal processing times an heuristic with a guaranteed performance ration and the given bound is tight.

6 Split graphs

We shall concentrate in this section on split graphs $G = (S, K; E)$ as compatibility graphs. The jobs S of G form a stable, hence are pairwise incompatible. The job set K forms a clique and can be placed into a single batch if the capacity is sufficient. Therefore, each job of S belongs to a separate batch, but may eventually be combined with some jobs of the clique K. For the following, we shall normalize the presentation of the split graph $G = (S \cup K; E)$: we assume that the clique K is maximal, i.e. if an element of S is connected to all nodes of K, we add this element to K.

We consider batch scheduling problems with split compatibility graphs of the form $B1/G = (S \cup K; E), b, \ldots /C_{max}$. According to the previous discussion, the values of the capacities b to study are : b variable and $b = k \in [3, |K| - 1]$ for finite capacities; $b = k \geq |K|$, no restriction on the clique size, in this case we set $b = \infty$.

For the static case, some previous works was established in [7, 8]. Table 7 summarizes the main problem types and their complexity status.

$p_S = p$ indicates the processing time of the jobs of the stable S are all equal to the constant p.

Table 7. Summary of previous results for split graphs

Problem	Complexity
$B1/G = (S \cup K; E), b = 2/C_{max}$	$O(n^{2.5})$
$B1/G = (S \cup K; E), b = k/C_{max}$ with $k \geq 3$	NP-hard
$B1/G = (S \cup K; E), b = \infty/C_{max}$ with $p_S = p$ and $\lvert\{J_j \in K \ / \ (i,j) \in E \ and \ p_j > p\}_{J_i \in S}\rvert \leq 2$	$\leq O(n^2 logn)$
$B1/G = (S \cup K; E), b, p_i = 1/C_{max}$	$\leq O(n^3)$
$B1/G = (S \cup K; E), b = \infty, p_i = 1/C_{max}$	$O(n)$

The special cases with split compatibility graph and release times have been analyzed in [4]. In this paper, each job has a release date and a processing time equal to 1. The dynamic case is then investigated. In this case, polynomial time solution algorithms was exploited, to solve some sub-problems of this NP-hard problem. In table 8 below the problem types and their complexity status are shown.

Table 8. Summary of results for split graphs

Problem	Complexity
$B1/G = (S \cup K; E), b = k, r_i, p_i = 1/C_{max}$ with $k \geq 2$	NP-hard
$B1/G = (S \cup K; E), b, r_S, r_K = 0, p_i = 1/C_{max}$	$\leq O(n^3 log\frac{n}{b})$
$B1/G = (S \cup K; S \times K), b, r_i, p_i = 1/C_{max}$	$O(nlogn)$

$r_S, r_K = 0$ indicates that the release times for the jobs of the stable S are arbitrary and the release times for the jobs of the clique K are all equal to 0. $B1/G = (S \cup K; S \times K)$ indicates a split complete graph.

7 Complementary of bipartite graphs

The complementary of bipartite graph is denoted by $G = (K_1 \cup K_2; E)$ and is formed of two cliques as subgraphs. The table 9 shows below the main results in this domain.

Table 9. Summary of results for complementary of bipartite graphs

Problem	Complexity	Reference
$B1/G = (K_1 \cup K_2; E), b \geq n/C_{max}$	NP-hard	[14]
$B1/G = (K_1 \cup K_2; E), b \geq n, p_i \in \{1, t\}/C_{max}$	$O(n^2)$	[14]
$B1/G = (K_1 \cup K_2; E), b \geq n, r_i, p_i = 1/C_{max}$	NP-hard	[3]
$B1/G = (K_1 \cup K_2; E), b \geq n, r_i \in \{0, r\}, p_i = p/C_{max}$	$O(n^2)$	[3]

8 Interval graphs

Batch scheduling problems with interval graphs have been studied in [18] according to some natural practical conditions that are presented. Some results are presented in table 10.

Table 10. Polynomial problems with interval graphs

Type	Complexity
$B1/G = INT, b, p_i = 1/C_{max}$	$O(n^2)$
$B1/G = INT, b \geq n/C_{max}$	$O(n^3)$
$B1/G = INT, b = k/C_{max}$ $(k \geq 3)$	open

9 Other graphs

Others graphs are also studied. Table 11 summarizes the problem types and their complexity status.

Table 11. Other summary of results

Type	Complexity	Reference
$B1/G = (V, E), b \geq n, p_i = 1/C_{max}$		[10]
where G is:		
circular-arc graph	$O(n^3)$	
chordal graph	$O(n^2)$	
comparability graph	$O(n^3)$	
$B1/G = (V, E), b \geq n/C_{max}$		[14]
where G is the complementary of:		
- line-graph L(H) of regular bipartite multi-graph with $\Delta(H) = 3$	NP-hard	
- cubic bipartite graph	NP-hard	
- k-regular bipartite graph	NP-hard	

10 Conclusion

In this paper, we have established new complexity results for the problem of minimizing the makespan on a single batch processing machine, in which jobs are not all compatible. This is indicated by a compatibility graph, which is assumed to be general, split, bipartite, complementary of bipartite, of intervals, etc. This last problem coming from a concrete application in the sheet metal

industry. For these problems, both the static case and the dynamic case are investigated.

Further extensions, arising in connection with the first application, may be defined. On one hand, it may be more appropriate to replace C_{max} with a flow time criterion, since one would like to reduce the storage of the semi-finished products on the shop floor. On the other hand, the compatibility relations may also be extended. Rather than only considering compatibility with respect to the height of the metal coils, one might also add compatibility for the diameters, weights, etc. each characteristic (measurement) has its tolerance limits and therefore defines an interval graph. this leads to a compatibility graph which is the intersection of interval graphs. such a graph may neither be an interval graph, nor even a perfect graph.

References

1. Ahmadi, J.H., Ahmadi R.H., Dasu S. and Tang C.S. : Batching and scheduling jobs on batch and discrete processors, *Operations Research* **40** (1992), 750–763.
2. Boudhar, M. : Ordonnancement sur machines à traitement par batch sous contraintes de compatibilité de tâches : complexité et approches algorithmiques, *Thèse de doctorat d'état, Université USTHB-Alger* (2004).
3. Boudhar, M. and Khelladi, A. : Ordonnancement par batchs sous contraintes de compatibilité de tâches, *Maghreb Mathematical Review* (2004), accepted, in press.
4. Boudhar, M. : Dynamic scheduling on a single batch processing machine with split compatibility graphs, *Journal of Mathematical Modelling and Algorithms* **2** (2003), 17–35.
5. Boudhar, M. : Scheduling a batch processing machine with bipartite compatibility graphs, *Mathematical Methods of Operations Research* **57** (2003), 513–527.
6. Boudhar, M. and Finke, G. : Problème d'ordonnancement de tâches sur une machine à traitement par batch, *Maghreb Mathematical Review* **10** (2001), 161–178.
7. Boudhar, M. : Static scheduling on a single batch processing machine with split compatibility graphs, *Cahiers du Laboratoire Leibniz-IMAGrenoble (France)* **28**, 2001.
8. Boudhar, M. : Scheduling on a batch processing machine with split compatibility graphs. *Journal of Mathematical Modelling and Algorithms* **4(4)** (2005), 391–407.
9. Boudhar, M. and Finke, G. : Scheduling on batch processing machines with constraints of compatibility between jobs, *Proceedings Second IFAC/IFIP/IEEE Conference on Management and Control of Production and Logistics (MCPL'2000)* **2** (Grenoble, France, 2000), pp. 703–708.
10. Boudhar, M. and Finke, G. : Scheduling on a batch machine with job compatibilities, *Belgian Journal of Operations Research, Statistics and Computer Science (JORBEL)* **40** (2000), 69–80.
11. Brauner, N., Dhaenens-Flipo, C., Espinouse, M.L., Finke, G. and Gavranovic H. : Decomposition into parallel work phases with application to the sheet metal industry, *Proceedings International Conference on Industrial Engineering and Production Management (IEPM'99)* **1** (1999), pp. 389–396.
12. Brucker, P., Gladky, A., Hoogeveen, H., Kovalyov, MY., Potts C., Tautenhahn, T. and Van De Velde, S. : Scheduling a batching machine, *Journal of Scheduling* **1** (1998), 31–54.

13. Chandru, V., Lee, C.Y. and Uzsoy, R. : Minimizing total completion time on batch processing machines, *International Journal of Production Research* **31** (1993), 2097–2121.
14. Demange, M., Werra, D., Monnot, J. and Paschos, V.T. : Weighted node coloring: when stable sets are expensive (extended abstract), *In L. Kucera (Ed.) : WG 2002, LNCS 2573, Springer-Verlag Berlin Heidelberg* (2002), 114–125.
15. Dobson, G. and Nambinadom, R.S. : The batch loading and scheduling problem, Research Report, Simon School of Business Administration, University of Rochester NY, 1992.
16. Escoffier, B., Monnot, J. and Paschos, V.T. : Weighted coloring: further complexity and approximability results, *proceedings of the conference ICTCS* (2005).
17. Finke, G. and Gavranovic, H. : Graph partitioning and set covering for the optimal design of a production system in the metal industry, *In Proceedings Second Conference IFAC/IFIP/IEEE on Management and Control of Production and Logistics (MCPL'2000)* **2** (Grenoble, France, 2000), pp. 603–608.
18. Finke, G., Jost V., Queyranne M. and Sebo A. : Batch processing with interval graph compatibilities between tasks, *Cahiers du Laboratoire Leibniz-IMAGrenoble (France)* **108**, 2004.
19. Garey, M.R. and Johnson, D.S. : *Computers and intractability: a guide to the theory of NP-Completeness*, W.H. Freeman and Company: San Francisco, 1979.
20. Hochbaum, D.S. and Landy, D. : Algorithms and heuristics for scheduling semiconductor burn-in operations, Research Report ESRC 94-8, University of California, Berkeley, USA, 1994.
21. Ikura, Y. and Gimple, M. : Efficient scheduling algorithms for a single batch processing machine, *Operations Research Letters* **5** (1986), 61–65.
22. Kubiak, W. and Jolai Ghazvini, F. : Minimizing earliness/tardiness criteria on a batch processing machine with job families, *Proceedings Second Annual International Conference on Industrial Engineering* **2** (1997), pp. 785–790.
23. Lee, C.Y. and Uzsoy, R. : Minimizing makespan on a single batch processing machine with dynamic job arrivals, *International Journal of Production Research* **17** (1999), 219–236.
24. Lee, C.L., Uzsoy, R. and Martin-Vega, L.A. : Efficient algorithms for scheduling semiconductor burn-in operations, *Operations Research* **40** (1992), 764–775.
25. Li, C.L. and Lee, C.Y. : Scheduling with agreeable release and due dates on a batch processing machine, *European Journal of Operational Research* **96** (1997), 564–569.
26. Mehta, S.V. and Uzsoy, R. : Minimizing total tardiness on a batch processing machine with incompatible job families, *IIE Transaction on Scheduling and Logistics* **31** (1998), 165–178.
27. Nguyen, P.L. : Planification tactique de la production : approche hiérarchisée pour une classe d'entreprises de sous-traitance et application au cas de laminage à froid, *Thèse de doctorat, Université Joseph Fourier, Grenoble*, 1997.
28. Potts, C.N. and Kovalyov, Y.K. : Scheduling with batching: A review, *European Journal of Operational Research* **120** (2000), 228–249.
29. Tarjan, R.E. : *Data structures and networks algorithms*, SIAM, 1983.
30. Uzsoy, R. : Scheduling batch processing machines with incompatible job families. *International Journal of Production Research* **33** (1995), 2685–2708.

Application des techniques de fouille de données à l'analyse des fichiers log
(Cas des serveurs proxy de l'UATL⋆)

B. ZIANI* and Y. OUINTEN**
{bziani, ouinteny}@mail.lagh-univ.dz

*,** Dépt. de génie-informatique, université de Laghouat 03000, Algérie

Abstract. Dans cet article nous nous intéréssons à l'analyse du comportement des utilisateurs de la connexion Internet de l'UATL. Cette connexion est assurée par le CERIST[1] et fait partie du projet ARN[2]. En appliquant certaines techniques de fouille de données, notre analyse a porté sur deux aspects. D'une part une étude des navigations des utilisateurs en terme de profils des sites visités, et d'autre part une classification (clustering) des utilisateurs selon certains critres retenus. Nous avons préviligié une approche expérimentale, plutôt qu'une approche par simulation afin de produire une analyse plus réaliste du trafic Web étudié. L'analyse est réalisée à l'aide de deux outils. WebSpy Analyzer pour une analyse exploratoire des fichiers log et WEKA (logiciel de fouille de données) pour réaliser une classification (clustering) des utilisateurs basée sur l'algorithme Kmeans.
Les résultats de l'expérience montrent une grande diversité des sites visités. Le clustering réalisé n'a pas donné lieu à des groupes qui se détachent visiblement les uns des autres. On en déduit un comportement similaire chez l'ensemble des utilisateurs, indépendemment de leurs profils (enseignant, étudiant, personnel administratif).
Mots clés: Extraction des connaissances à partir des données, fouille de données(Data mining), fouille de Web (Web mining), fichier log, clustering.

1 Introduction

La fouille de données constitue le cœur du processus de l'ECD[3]. Sa motivation principale est la valorisation d'une grande base ou entrepôt de données ***(data werhouse)*** par la recherche de connaissances nouvelles pour l'aide à la décision[3].
Compte tenu de la croissance exponentielle de l'Internet, les applications des techniques de la fouille de données aux fichiers log sont rapidement apparues[9].

⋆ Université Amar Télidji de Laghouat
[1] CEntre de Recherche sur l'Information Scientifique et Technique -www.cerist.dz-
[2] Academic Research Network - www.arn.dz -
[3] ECD=Extraction des Connaissances à partir des Données

Dans ce cas, où les données analysées proviennent du Web, on parle plutôt de *Web mining*. Selon le type des données analysées, on distingue trois classes d'applications dans le domaine du Web mining:

• **Web Content Mining:** qui concerne l'analyse du contenu des pages Web.
• **Web Structure Mining:** qui s'intéresse à l'analyse de la structure des sites Web.
• **Web Usage Mining:** qui analyse le comportement des utilisateurs des sites Web.

Dans notre travail on s'intéresse à une application du Web Usage Mining. Ce dernier peut être définit comme étant "l'application des techniques de fouille de données pour découvrir des comportements typiques d'utilisation dans le but de mieux comprendre et servir les utilisateurs du Web"[9]. Ce processus d'analyse comporte trois étapes principales:

1. Le prétraitement: préparation des données à analyser.
2. Extraction des modèles: découverte des comportements typiques à partir des données analysées.
3. Interprétation: analyse et validation des modèles découverts.

2 Sources de données

Les principales sources de données pour le processus du Web Usage Mining sont les postes **clients**, les **serveurs** Web, et les serveurs **proxy**.

• Les données des postes client
Ces données proviennent du fait que souvent le poste client réserve un espace disque de la machine pour pouvoir stocker une quantité de données en cache. Ce processus permet d'accélérer la navigation en ne redemandant au serveur que les pages nouvellement visitées. Afin d'exploiter cette source de données des outils de recueil de trafic doivent être installés. Ces outils (des applets Java en général) sont des programmes insérés afin d'être chargés et exécutés par le navigateur. Aussi, lorsque le navigateur charge le fichier demandé par le client, il exécute le programme et génère ainsi une ligne dans le fichier log du serveur sollicité. Cependant cette méthode présente quelques inconvénients. D'une part Elle alourdi la navigation et d'autre part les utilisateurs ont la possibilité d'interdire au navigateur d'exécuter les applets Java. Ceci rend difficile l'analyse et l'exploitation de ces données.

• Les données des serveurs Web
Les serveurs Web constituent une importante source de données sur le net. En effet, les requêtes des utilisateurs sont stockées dans les fichiers log de ces serveurs. La taille de ces fichiers peut atteindre plusieurs centaines de Mégaoctets par jour. Techniquement l'utilisateur demande une page au serveur Web qui héberge le site. Le serveur au cours du procéssus de réponse la demande va stocker dans un

fichier log, au format texte, les paramètres de la demande. Le fichier log contient principalement:

- L'adresse IP de l'utilisateur qui a fait la requête
- La date et l'heure de la requête
- Le protocole de la requête (http, https, ftp,...)
- La requête elle-même : /dossier/fichier.html
- Le résultat du traitement de la demande : Par un code: 200 pour élément fourni, 404 page non disponible, etc.
- La taille de l'élément envoyé
- La signature du navigateur : Cette information permet d'identifier la navigateur et le système d'exploitation.

• **Les données des serveurs proxy**
Les serveurs proxy ont vu le jour aussi bien pour des raisons de sécurité que de trafic réseau. Un serveur proxy peut être définit comme étant *une machine intermédiaire placée entre le client et le serveur, qui permet d'acheminer indirectement les requêtes de l'un vers l'autre.* Pour toute requête que le proxy reçoit, il la lit, la traduit, l'envoie au serveur et renvoie la réponse de ce dernier au client émetteur. L'intérêt est donc multiple: dans un réseau local, une seule machine communique avec l'extérieur et ainsi la sécurité est plus facile à gérer, de plus le principe d'intermédiaire centralisé permet une mise en place globale de règles de sécurité (accès interdit à certains domaines, ports de connections bloquées...). Les fichiers log de tels serveurs contiennent les mêmes informations que ceux des serveurs Web. Ils stockent les accès de différents utilisateurs à différents sites. Ils sont utilisés pour identifier les profiles de navigation d'un groupe d'utilisateur partageant le même serveur proxy.

3 Objectifs

L'objectif de notre travail est d'analyser le comportement des utilisateurs de la connexion Internet de l'UATL. Il s'agit de chercher à mettre en évidence des corrélations entre l'usage de la connexion et des groupes d'utilisateurs afin de valider (ou d'invalider) l'existence de comportements particuliers chez les utilisateurs suivant leur profil à savoir **enseignant**, **étudiant**, ou **personnel administratif**. Les inérêts sont multiples:
• Obtenir une visibilité et des informations sur l'exploitation réelle du réseau.
• Anticiper des besoins futurs du réseau en terme de bande passante et de sécurité.
• Améliorer les services offerts en procédant à une gestion des serveurs proxy par groupe et par profil d'utilisateurs.

4 Les outils utilisés

Tous les outils d'analyse des fichiers log consultés ne permettent qu'une étude exploratoire des ces fichiers. Ils n'intègrent pas d'autres techniques de fouille de

données, en particulier le clustering qui constitue un volet important de notre étude. D'autre part, les outils de fouille de données ne reconnaissent pas le format des fichiers log. On a été contraint à choisir un outil d'analyse des fichiers log associé à un autre de fouille de données. Une solution qui semble offrir plus de souplesse pour l'analyse. Ainsi nous avons réalisé notre analyse à l'aide de deux outils: WebSpy Analyzer (version Giga 2.1.3.15) pour une analyse exploratoire des fichiers log et WEKA (logiciel de fouille de données, version 3.4.7) pour réaliser une classification (clustering) des utilisateurs basée sur l'algorithme Kmeans.

• **WebSpy Analyzer**
WebSpy Analyzer est un produit d'analyse et de reporting Internet et Email de la société WebSpy[4]. Il permet d'analyser les fichiers log des pare-feux, des serveurs proxy et des serveurs de messagerie. Il supporte plus de 60 formats de fichiers log et crée des rapports sur la navigation des utilisateurs. Il est recommandé que Analyzer Giga soit installé sur un ordinateur PIII à 500 MHz ou plus, sous Windows XP , ayant un système de fichiers NTFS, et au moins 256 Mo de RAM. Il faut disposer d'Internet Explorer 5.5 ou plus pour obtenir des rapports en format HTML.

• **Weka**
Weka (Waikato Environment fot Knowledge Analysis) est un ensemble de classes et d'algorithmes en Java implémentant les principaux algorithmes de fouille de données. Il est disponible en open source sur le Web[5]. Weka travaille avec des fichiers de type ".arff" (attribute relation file format) qui représentent les fichiers de données à analyser. Ces fichiers sont composés d'une **entête** décrivant le fichier, suivi de la liste des **attributs** et leur type et en fin l'ensemble des différentes **valeurs**.

5 Etude expérimentale

5.1 Description

Les serveurs proxy de l'UATL enregistrent l'ensemble de l'activité des utilisateurs de la connexion Internet. On y trouve la trace de toutes les requêtes effectuées par tous les postes clients dépendant des serveurs en question. On peut ainsi savoir comment chacun utilise le Web dans l'établissement.
A partir des données brutes de trafic, l'étude expérimentale vise, dans un premier temps, une analyse exploratoire de l'activité des utilisateurs. Les résultats obtenus sont ensuite exploités afin de décrire les utilisateurs par différents attributs tels que le temps de navigation, le temps de téléchargement et le volume des données échangées. Ces attributs sont utilisés pour réaliser un regroupement (clustering) des utilisateurs. La partie expérimentation s'est déroulée en trois étapes:

[4] www.webspy.com
[5] www.cs.waikato.ac.nz/ml/weka

1. Préparation des données : Récupération et concaténation des fichiers log des différents serveurs proxy de l'UATL sur la période d'une semaine.
2. Analyse de l'activité : cette partie de l'analyse s'articule sur l'identification du comportement des utilisateurs d'une manière générale : intensité d'usage, services utilisés, profils des sites visités, fréquences des visites, etc. Cette partie de l'analyse est réalisée à l'aide de l'outil WebSpy Analyzer.
3. Regroupement (clustering) des utilisateurs : A partir des résultats obtenus dans l'étape précédente, une segmentation des utilisateurs est réalisée à l'aide de l'outil Weka. L'objectif étant l'identification d'éventuels groupes homogènes d'utilisateurs.

5.2 Préparation des données

En collaboration avec les responsables du centre de calcul, nous avons pu obtenir un jeu de données sur la période du 02 au 08 Mai 2006. Les différents fichiers log sont concaténés quotidièmement. A la fin de la semaine tous les fichiers sont concaténés une nouvelle fois en un seul fichier qui va constituer notre source de données. Le tableau 1 indique la taille des différents fichiers log récupérés sur la période de l'expérience.

Jour	**Taille du log(en Mo)**
02/05 (Mardi)	227.2
03/05 (Mercredi)	193.2
04/05 (Jeudi)	73.2
05/05 (vendredi)	8.6
06/05 (Samedi)	186.5
07/05 (Dimanche)	199.9
08/05 (Lundi)	203.2
Moyenne	155.9

Table 1. Taille des fichies log analysés

5.3 Analyse de l'activité

L'objectif de cette première partie de l'étude expérimentale est de savoir comment chacun utilise le Web dans l'établissement. Le jeu de données analysé a une taille 1.08Go. Il contient 458 adresses IP différentes (458 utilisateurs) et 6073486 hits[6]. Le rapport généré à l'aide du logiciel WebSpy Analyzer présente une exploration détaillée des données (près de 36 pages). Dans ce qui suit nous nous intéréssons aux résultats répondant à nos besoins à savoir les sites et les profils visités par les utilisateurs.

[6] hit=toute requête faite à un serveur à partir d'un navigateur

• **Les sites les plus visités**
Le tableau 2 présente les 20 sites les plus visités par l'ensemble des utilisateurs.

Num.	Sites	Hit
01	221.7.12.210	209567
02	us.i1.yimg.com	173059
03	eur.i1.yimg.com	109622
04	www.google.fr	93073
05	www.google.com	89989
06	www.aljazeera.net	82295
07	kh.google.com	76297
08	audio.islamweb.net	52743
09	utils.winantivirus.com	52126
10	freempegs.aebn.net	43836
11	www.bentelhalal.com	42054
12	www.lagh-univ.dz	41929
13	row.bc.yahoo.com	38475
14	www.01net.com	35536
15	eur.a1.yimg.com	35220
16	images.maktoob.com	34827
17	38.118.213.207	28258
18	images.google.fr	24520
19	adsnew.maktoob.com	22931
20	38.118.213.215	22608

Table 2. Les sites les plus visités

• **Discussion**
Dans l'ensemble les tendances des utilisateurs sont très hétérogènes. Des outils de recherche et sites de téléchargement aux annuaires d'information en passant par les sites à caractère social et de divertissement. Il faut noter l'existence de sites dont le domaine (et donc le profil) n'a pu être identifié, même en utilisant un résolveur DNS[7]. Ils sont représentés par leurs adresses IP (telle que 221.7.12.210) et non par leurs nom d'URL. Ce sont des sites qui naissent et disparaissent rapidement et qui sont utilisés, généralement, pour un partage illégal de fichiers.

• **Les profils visités**
Les profils sont une manière simple de grouper (par un ensemble de mots clés) les sites pour une meilleure analyse. Ils permettent de créer des catégories de navigations à partir des fichiers log. WebSpy Analyzer a un ensemble de profils par défaut, comme Publicité ('Advertising') et Technologies de l'Information

[7] DNS = Domain Name System, est un système permettant d'établir une correspondance entre une adresse IP et un nom de domaine

('Information Technology'). Chaque navigation ne correspondant pas à un profil défini est classée dans le profil Divers ('Miscellaneous'). WebSpy Analyzer permet d'enrichir l'ensemble des mots clés des profiles existants ou de définir carrément de nouveaux profils. En plus des profils existants, nous avons définie deux nouveaux profils. (**Enseignement, science et éducation**) faisant référence aux sites à caractère scientifique et (**religion**) définissant les sites à caractère religieux. Le tableau 3 présente les différents profils visités par les utilisateurs.

Num.	Profiles	Hits	%
01	Entertainment	1324643	21.81
02	Miscellaneous	1081603	17.80
03	Information Technology	836397	13.77
04	Search Engines	642998	10.59
05	Enseignement, science et éducation	422782	06.96
06	Social and Chat	415105	06.83
07	News and Reference	408434	06.72
08	Advertising	292841	04.82
09	Personal Finance and Shopping	236857	03.90
10	Religion	174360	02.87
11	Adult	158078	02.60
12	My Organization	71841	01.18
13	Career	4507	00.07
14	Gambling	3040	00.05

Table 3. Les profils visités par les utilisateurs

• **Discussion**

La remarque principale à la lecture de ces résultats est le taux élevé (près de 18%) des sites visités non classés dans un profil connu. Ils sont ainsi regroupés dans la rubrique divers. Au début de l'expérimentation le taux était beaucoup plus important (près de 30 %). Ceci est du à la manière dont l'outil WebSpy Analyzer détermine le profil du site visité. En effet WebSpy Analyzer analyse les mots contenus dans l'URL du site. Ce dernier est ainsi affecté au profil contenant les mots clés identifiés dans son URL. Dans beaucoup de cas l'URL du site ne comporte pas des mots clés permettant de définir son profil. par exemple le site www.b11b.com ne peut être classé par simple analyse de son nom. Pour palier à ce problème nous proposons d'une part d'exploiter les mots clés figurant dans les pages HTML (les balises meta). D'autre part, quand ces mots clés ne sont pas disponibles, une visite du site devient nécéssaire pour définir son profil. La définition des profils, qui est une étape importante de la phase de préparation des données, devient ainsi une procédure semi-automatique. D'autre part les résultats montrent une tendance ne reflétant pas la vocation de l'établissement. Les sites à caractère scientifique ne représente qu'environ 7% du total de requêtes

contre 22% pour des sites de loisirs. Le site de l'établissement, quand à lui, représente environ 1% du total des visites éffectuées par les utilisateurs.

6 Clustering des utilisateurs avec Weka

6.1 Introduction

L'expérience de clustering a pour but de valider ou d'invalider l'existence d'un comportement typique des utilisateurs selon leur profil à savoir **enseignant**, **étudiant** ou **personnel administratif**. Les différents clusters qui peuvent être éventuellement identifiés présentent chacun un intérêt pour une catégorie différente d'utilisaeurs. Ceci permet un bonne vision de l'ensemble d'utilisateurs. Le clustering est réalisé à l'aide de l'algorithme **Kmeans**.

6.2 L'algorithme Kmeans

Proposé en 1967 par MacQueen [6], l'algorithme des centres mobiles (K-means) est l'algorithme de clustering le plus connu et le plus utilisé, tout en étant très efficace et simple. Ce succès est d au fait que cet algorithme présente un rapport cot/efficacité avantageux[2]
On considère l'espace de (n) points de dimension (p) suivant:

$$\mathbf{X} = \begin{bmatrix} x_1^1 \ \ldots \ x_1^j \ \ldots \ x_1^p \\ \ldots\ldots\ldots\ldots\ldots \\ x_i^1 \ \ldots \ x_i^j \ \ldots \ x_i^p \\ \ldots\ldots\ldots\ldots\ldots \\ x_n^1 \ \ldots \ x_n^j \ \ldots \ x_n^p \end{bmatrix}$$

On suppose que les (n) points peuvent être regroupés en (c) clusters $(c < n)$. les clusters sont décrits par leurs centres:

$$\mathbf{C_k} = \left[x_k^1, x_k^2, \ldots, x_k^j, \ldots, x_k^p \right]$$

L'algorithme Kmeans fonctionne selon les étapes suivantes:

1. On choisit d'abord K points représentant les K clusters recherchés et appelés "centroides" des clusters (le nombre K est fixé par l'utilisateur).
2. On assigne chaque point non classé au cluster dont le centroide est le plus proche.
3. On réevalue les nouvaux centroides des clusters.
4. On recommence (réassignation des points au cluster dont le centroide est le plus proche, et recalcul des centroides) jusqu'a ce que les centroides ne changent plus significativement.

6.3 Préparation des données

Pour réaliser ce clustering, nous avons éffectuer une sélection des attributs jugés pertinents. En deuxième lieu nous avons généré un fichier, au format **arff** (format reconnu par Weka) , contenant toutes les informations nécéssaires. Le jeu de données utilisé dans l'expérience de clustering concerne un ensemble de 458 utilisateurs décrits par 7 attributs. Le tableau 4 représente les attribus retenus pour le processus de clustering.

Attribut	Type	Signification
Nom	String	Nom désignant un tilisateur et remplaant son adresse IP (pour confidentialité)
Tdown	Real	Temps épuisé pour des téléchargements
Tnav	Real	Temps épuisé pour des navigations
In	Real	Volume de données entrant
Out	Real	Volume de données sortant
Pv1	Enumerate	Le premier profil le plus visité par un utilisateur
Pv2	Enumerate	Le deuxième profil le plus visité par un utilisateur
Profil	Enumerate	Le profil de l'utilisateur (Ens=Enseignant, Etd=Etudiant, Adm=personnel administartif)

Table 4. Les attributs retenus pour le processus de clustering

La préparaion des données au format **.arff** a nécéssité un travail laborieux. Dans cette phase de l'expérimentation nous avons procédé principalement à :

- La codification des adresses IP des utilisateurs (pour confidentialité)
- La codification des profils visités par les utilisateurs pour faciliter leur manipulation en type enuméré
- La conversion des attributs **Tnav** et **Tdown** de la forme **hh:mm:ss** en seconde pour faciliter leur manipulation sous forme numérique.

6.4 Clustering

Le clustering K-Means sous Weka nécéssite la determination de deux paramètres qui sont le nombre de clusters recherchés et une "graine". Cette notion de graine correspond à un point de départ aléatoire dans l'esapce pour calculer les différents clusters (initialisation des centroides). Selon la graine de départ choisie les points représentant les clusters vont évoluer jusqu'à un état de stabilité qui conduira à la fin du clustering.

L'objectif de l'expérience de clustering étant de confirmer (ou d'infirmer) l'existence de comportement typique selon le profil d'utilisateur, le nombre de clusters est ainsi fixé à trois (03). Par plusieurs expérimentations nous avons constaté que la variation de la graine (Seed) générait des clusters différents. Une fois le résultat

obtenu par l'algorithme, une interprétation, souvent laborieuse, est nécessaire. En sortie de chaque expérimentation, le logiciel nous fourni les informations suivantes:

- Les centroides (centres des clusters)
- Le nombre d'instances assignées à chacun des clusters
- La distribution des instances dans les clusters
- Une estimation des clusters représentant le mieux les classes existantes
- Une estimation des instances mal classées

Notre intérêt est d'obtenir des clusters les plus significatifs possibles. Le "meilleur" résultat sera évalué en nous basant sur le taux d'erreurs (pourcentage des instances mal classées).

6.5 Résultats et discussion

Nous avons effectué le clustering pour différentes valeurs du paramètre d'initialisation aléatoire des centroides en faisant varier la valeur de la graine (Seed) de 0 à 20. Pour les différents résultats obtenus, le taux d'erreurs (les instances mal classées) varie entre **56.98%** et **63.97%**. Le résultat corréspondant au taux d'erreur le plus faible est celui relatif à une graine de **10**. Les résultats obtenus pour ce cas sont:

• **Pourcentage des instances dans les clusters :**

Cluster	Nombre des instances affectées au cluster
0	82 (18%)
1	185 (40%)
2	191 (42%)

• **La composition des clusters en fonction du profil des utilisateurs est :**

	0	1	2
Ens	12 (08.27%)	82 (56.55%)	51 (35.17%)
Etd	55 (31.43%)	40 (22.86%)	80 (45.71%)
Adm	15 (10.87%)	63 (45.65%)	60 (43.48%)

• Le nombre des instances mal classées est de 261 soit un pourcentage de 56.98%.

• **Discussion**
Le taux élevé des instances mal classées ainsi que la composition des clusters en fonction des profils des utilisateurs montrent bien que les classes trouvées ne se détachent pas visiblement les unes des autres. Ce résultat reflète un comportement assez homogène chez l'ensemble des utilisateurs indépendemment de leurs profils.

7 Conclusion et perspectives

L'enseignement principal du premier aspect de l'étude consiste en la mise en évidence d'une grande diversité des profils visités. Contrairement à ce qui était attendu, l'accès au Web ne porte pas sur des profils reflétant la vocation de l'établissement, mais plutôt sur un ensemble très hétérogène de sources d'informations. D'autre part l'échange avec d'autres établissement du projet ARN est totalement absent. Ceci reflète l'état du dit projet vis à vis des ses objectifs dont le plus ambitieux est l'exploitation de la connexion pour l'échange entre les universités Algériennes. Nous pensons que la validation des résultats obtenus nécéssite d'autres expérimentations dans le temps et, éventuellement, une comparaison avec des études similaires dans des institutions universitaires. La partie expérimentation nous a montré combien pouvait être laborieuse l'étape de préparation des données. Comme indiqué dans la littérature consultée, nous confirmons qu'elle représente de 60 à 80% du processus d'analyse.
L'expérience de clustering montre qu'il n'y a pas des groupes d'utilisateurs qui se détachent d'une manière apparente. Ainsi l'ensemble des utilisateurs peut être vu comme un seul groupe homogène. Il serait intéréssant de valider ce résultat par d'autres expérimentations. Un suite de travail pourrait s'orienter vers la prise en compte d'autres critères pour le clustering et ,éventuellement, des expérimentations avec d'autres algorithmes de clustering.

References

1. Mireille Arnoux, Yves Lechevallier, Doru Tanasa, Brigitte Trousse, and Rosanna Verde. Automatic clustering for the web usage mining. In Dana Petcu, Daniela Zaharie, Viorel Negru, and Tudor Jebelean, editors, *Proceedings of the 5th Intl. Workshop on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC03)*, pages 54 – 66. Editura Mirton, Timisoara, 1-4 October 2003.
2. Laurent Candillier. La classification non supervisée. Technical report, Equipe GRAppA, Lille 3, Sepembre 2004.
3. R. Rakotomalala D.A. Zighed. Extraction de connaissances partir de données (ecd). *Techniques de l'Ingénieur*, H 3 744, 2003.
4. Acha El Golli, Yves Lechevallier, Doru Tanasa, Brigitte Trousse, and Rosanna Verde. Classification automatique : Applications au web mining. In *Séminaire LISE-CEREMADE Paris-Dauphine*, 2 December 2004.
5. M. Deshpande J. Srivastava, R. Cooley and P. Tan. Web usage mining: Discovery and applications of usage patterns from web data. *SIGKDD explorations*, 1(2):12–23, January 2000.
6. J. MacQueen. Some methods for classification and analysis of multivariate observations. In *5th Berkeley symposium on Mathematical statistics and probability*, pages 281–297, Berkeley, 1967.
7. Christine Michel. Caractérisation d'usages et personnalisation d'un portail pédagogique : état de l'art et expérimentation de différentes méthodes d'analyse du web usage mining. In *7me colloque de l'AIM : "Affaire Electronique et Société de Savoir : Opportunités et Défis" (AIM.7)*, Hammamet, Tunisie, 30 Mai - 1er Juin 2002.

8. Doru Tanasa, Brigitte Trousse, and Florent Masseglia. Classer pour découvrir: une nouvelle méthode d'analyse du comportement de tous les utilisateurs d'un site web. In Georges Hébrail, Ludovic Lebart, and Jean-Marc Petit, editors, *Revue des Nouvelles Technologies de l'Information (RNTI), numéro spécial Extraction et Gestion des Connaissances (EGC'2004)*, volume 2, pages 549–560. Cépauds-Editions, January 2004.
9. Doru Tanasa, Brigitte Trousse, and Florent Masseglia. *Mesures de l'internet*, chapter Fouille de données appliquée aux logs web : état de l'art sur le Web Usage Mining, pages 126–143. Les Canadiens en Europe, 2004. Sous la direction d'Eric Guichard.

An Overview On Predictive And Preemptive Maintenance Techniques In Manets

Sofiane Boukli Hacene[1], Ahmed Lehireche[1]

[1] Evolutionary Engineering and Distributed Information Systems Laboratory, Computer Science Department, Sidi Bel Abbes University, Algeria
boukli@univ-sba.dz, elhir@yahoo.com

Abstract. Ad hoc networks (Manets) are characterized by multi-hop wireless connectivity, dynamic changing in network topology and the need for efficient dynamic routing protocols. The ephemeral nature of links between network nodes can cause routes to become unstable so unusable. A node on a route from a source to a destination may become unreachable from its predecessor node because of node movement. Routing protocols typically include mechanisms for route discovery and route maintenance. The performance of these protocols depends on the route maintenance mechanism they use. Several studies have been conducted [3], [4], [5], [7] [8], [9] to enhance route maintenance using either preemptive or predictive mechanisms. In those approaches route maintenance is initiated to anticipate link break rather than waiting for the break to happen. In this paper, we present different studies on predictive and preemptive approaches to route maintenance and we detail our implemented approach [11] that combine both predictive and preemptive, and gather there advantages.

Keywords: Ad Hoc Networks, Mobile Networking, Predictive Routing, Preemptive Routing, AODV.

1 Introduction

A mobile ad hoc network (Manet) consists of a set of mobile wireless devices that can communicate with each other without requiring the existence of fixed networking infrastructure. Nodes cooperate with their neighbors to route data packets to their final destinations. As nodes are mobile, routes between sources and destinations need to be determined and adjusted dynamically. Routing protocols for Manets typically consist of a route discovery mechanism and a route maintenance mechanism. The route discovery mechanism is invoked to determine a route between a sender and a receiver. The route maintenance mechanism finds new valid routes as substitutes for broken active routes. Several approaches to route maintenance have been proposed in the literature. In the Ad-hoc On-Demand Distance Vector (AODV) protocol, a node that detects the failure of a link to a destination propagates a link failure control packet to all upstream (towards sources) neighbors. This process is repeated until all sources that use the link are notified. Source nodes invoke the route discovery

mechanism if they need a route to the destination. Link failures can be detected via link-layer acknowledgements or periodic HELLO packets [1]. In addition to this global maintenance by the sources, a local mechanism is also added to AODV. A node that detects the failure of a link to a destination attempts to find a sub-path that can reach the destination [1], [2].

A problem with waiting for a link to fail before route maintenance is carried out is the time failure notification and route re-establishment operations. Schemes that use Global Positioning System (GPS) information for detecting and handling expected link failures early have been proposed [3], [4], and [5]. The power of received signals has also been used for detecting expected link failures in DSR [6]. The area covered by the transmission range of a signal is subdivided into a safe range and a preemptive range. When it is determined that a node is in the preemptive range, a link failure warning message is sent to the source of the packet. The source then finds and uses a substitute route. This approach can result in unnecessary warning messages and route maintenance operations because the link that is expected to break may not fail soon, depending on movement speed and direction.

Our proposed approach uses the Lagrange interpolation for estimating whether an active link to a neighboring node will fail. When link failure is expected between a node and an upstream neighbor, the upstream neighbor itself first attempts to find a route to the destination. If such route is not found within a discovery period a link failure warning is propagated via upstream nodes to sources that use this link. Source nodes invoke the route discovery mechanism if they need a route to the destination.

The remainder of the paper is organized as follows. In Section 2, we describe in some detail the route maintenance operation in a well known routing protocol in Manets. This is followed by defining preemptive and predictive route maintenance and classifying studies treating the subject in Section 3. In section 4, we present our approach that gathers the advantages of preemptive and predictive approaches of route maintenance. Finally, we conclude in Section 5.

2 Aodv Routing Protocol

AODV [1] is an on-demand routing protocol. Route discovery is initiated only when a source node needs to communicate with a destination for which it does not have a route in its routing table. To discover a route to a destination, the source node broadcasts a route request message (RREQ) that contains a request ID. If a node receives a RREQ that it has already received, it drops the request. Otherwise, it stores the address of the node from which it received the request so as to establish a reverse route to the source that it uses later. If the RREQ reaches a node that has a route to the destination, the node sends, over the reverse route, a route reply message (RREP) to the source. The reply message contains the number of hops needed to reach the destination from the node. If the RREQ reaches the destination, it sends a route reply to the source over the reverse route. Intermediate nodes that do not have a path to the

destination re-broadcast the request. As the RREP is sent back to the source over the reverse path each node stores the address of the node that sent the reply. The forward path thus determined from the source to the destination is used for sending packets to their destination. AODV uses sequence numbers maintained for the different destinations so to guarantee freshness of routing information.

A link breaks when a node within an active route moves out of the transmission range of its upstream neighbor. When a link break occurs, the node upstream the break invalidates, in its routing table, all routes become unusable due to the loss of the link. It then creates a Route Error (RERR) message, in which it lists the destinations that have become unreachable because of the loss of the link. The RERR is sent to all source nodes that use the link. This procedure is named *global repair*. AODV also includes a *local repair* mechanism to locally recover from link losses. Local repair is triggered when a link break occurs between nodes within an active route. In this repair, the node upstream the break tries to find alternative sub-paths to the destinations of packets that it has received, but is unable to forward them because of the link break.

3 Route Maintenance Mechanisms

Route failures have a significant negative impact on packet delivery. Packet dropping and higher delays are the main consequences of route failures. The time elapsed between link break detection and alternative path establishment can be high. Therefore, many studies have focused on improving route repair by adding preemptive or predictive mechanisms.

The idea behind preemptive route maintenance is to use techniques to look for a new route to a destination that the node already could have an active route to, before a route failure happens.

In predictive route maintenance we guess when a route failure will happen by using a prediction function and we repair this route if we are in need of it. This means that we prevent route failure to happen.

3.1 Predictive route maintenance

Su et al. [8] propose the inclusion of mobility information in data packets so as to estimate the expiration time of routes at the destinations in on-demand routing protocols. Predicted link expiration times are appended to data packets as they travel to their destinations. A Distance Vector with Mobility Prediction (DV-MP) routing protocol is also proposed. This protocol selects route links based on their predicted expiration times. Mobility prediction utilizes GPS location information. Su et al. predicted the value Link Expiration Time (LET) using the formula below:

$$LET_{i,j} = \frac{-(a \times b + c \times d) + \sqrt{(a^2 + c^2) \times r^2 - (a \times d - b \times c)}}{a^2 + c^2} \quad (\mathbf{1})$$

Where:

$$a = v_i \cos\theta_i - v_j \cos\theta_j \quad (\mathbf{2})$$

$$b = x_i - x_j \quad (\mathbf{3})$$

$$c = v_i \sin\theta_i - v_j \sin\theta_j \quad (\mathbf{4})$$

$$d = y_i - y_j \quad (\mathbf{5})$$

(x_i ,y_i) are the coordinate of mobile host *i* and (x_j ,y_j) are that of mobile host *j* where each mobile host is within the transmission range *r* of each other. v_i and v_j are the speeds, and θ_i and θ_j ($0 \le \theta_i$, $\theta_j < 2\pi$) are the moving directions of nodes *i* and *j*, respectively.

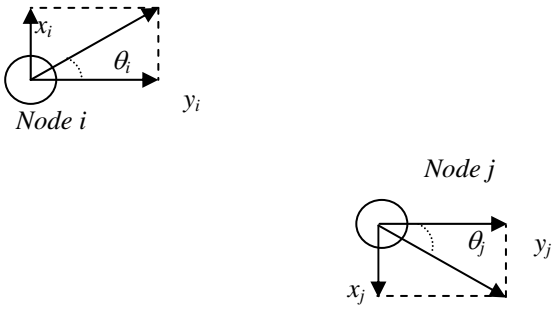


Fig. 1. Mobile nodes moving at random speeds and directions.

Its general goal is enhancing the performance of unicast and multicast routing protocols by selecting stable routes and performing route reconstruction proactively. An issue with global positioning systems is the cost associated with their use. A second issue is that they may not work properly because of fading.

Cahill et al. [3] propose the use of node position and mobility information in the route discovery mechanism of DSR. When multiple routes to a destination exist, route selection is based on route stability and hop count. Moreover, nodes upstream of links

that are predicted to break carry out preemptive local repair. This proposal uses DSR caching [6]. Therefore, it is not applicable to AODV. Moreover, a GPS is used.

In [4], Crisòstomo et al. propose a Preemptive Local Route Repair (PLRR) extension to AODV. Nodes trigger the preemptive local route repair procedure when they predict that a link on the route to a destination is about to break. All packets are modified so as to contain node positions and motion information obtained using GPS receivers that nodes are equipped with. The prediction function adopted in this work is an extension of the function proposed in Su et al. [8]:

$$LET_{i,j} = \frac{-(a \times b + c \times d) + \sqrt{(a^2 + c^2) \times (r - errpos_i - errpos_j)^2 - (a \times d - b \times c)}}{a^2 + c^2} \quad (\mathbf{6})$$

Where $errpos_i$ and $errpos_j$ are the maximum position error of neighbors i and j to account for position inaccuracies.

The problems with this approach are the cost associated with using a GPS and the need for synchronization between the internal clocks of nodes.

3.2 Preemptive route maintenance

In [5], Goff et al. propose a preemptive route maintenance extension to on-demand routing protocols. Its aim is to find an alternative path before the cost of a link failure is incurred. The received transmission power is used to estimate when a link is expected to break. A link is considered likely to break when the power of the signal received over it is close to the minimum detectable power.

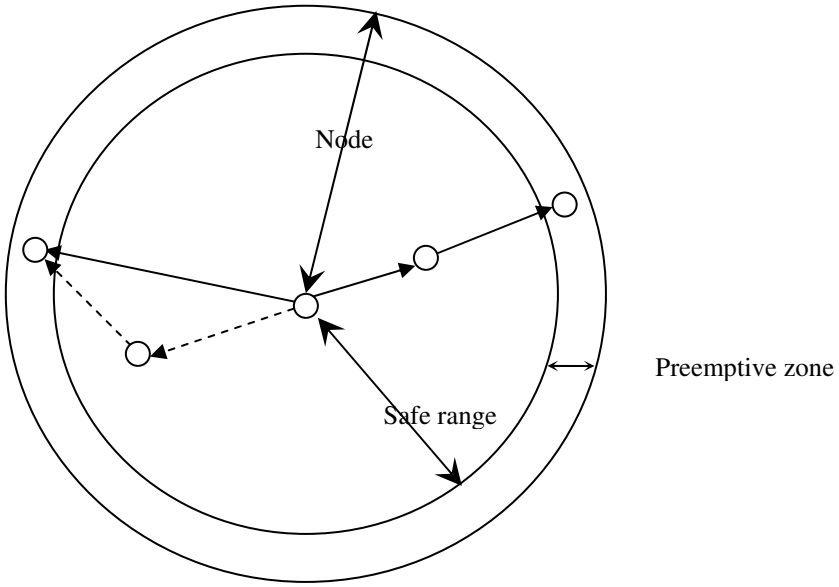


Fig. 2. Preemptive Zone

Route repair is the responsibility of a source node after receiving a warning about the imminence of a link break on an active route to a destination. This mechanism has been applied to DSR; AODV is also considered, but only superficially.

In [7], Srinath et al. propose a preemptive router handoff strategy based on received signal strength for AODV. Whenever a node *X* predicts that the previous or next link on an active path to a destination *D* will soon (within 0.25 seconds) have a signal strength below a threshold value, it attempts to find a neighbor *H,* that can reach both the predecessor *P* and successor *N* nodes with links that have superior strengths. The sub-path *P-X-N* is replaced with *P-H-N*, except when *H* has a better route to *D* that does not go through *N*. Predicted power strengths are based on the rate at which power changes between active neighboring nodes.

In [9], Boukerche and Zhang propose a Preemptive AODV (PrAODV). They combine two pre-emptive mechanisms, *Schedule a Rediscovery in Advance* and *Warn the Source Before the Path Breaks*. *Warn the Source Before the Path Breaks* is attained by exchanging Hello Message (ping–pong process). When the signal power of received packets is below a threshold value, the node sends to its upstream neighbors a hello message called "ping", neighboring nodes should reply with a hello packet called "pong" within a timeout. If the node doesn't receive this pong message a warning message should be sent back to the source. Upon receiving a warning message, the source node starts a rediscovery. *Schedule a Rediscovery in Advance* is achieved by collecting information of the active links. When a reply packet returns to the source through an intermediate node, it collects the information of the links.

Therefore, when the packet arrives at the source, the information about the condition of all links, including the minimum value of the lifetime of the links, will be known. Hence, we can schedule a rediscovery $T_{rediscovery}$ time before the path breaks. An issue with previous techniques is the high routing cost (overhead), however, the average delay end to end delay per packet, in PrAODV, is significantly reduced.

4 Predictive Preemptive route maintenance [11]

Predictive route maintenance techniques suffer of excessive computing and preemptive once generates false warning. In order to avoid those problems we proposed an approach that combines the two techniques.

Our approach (PPAODV) [11] is studied by extending the Ad-hoc On-Demand Distance Vector (AODV) protocol [1], [2]; however, it is general and can be used to enhance other routing protocols. The main goal of our approach is to avoid sending unnecessary warning messages. In this work, we consider that a node is in an unsafe or preemptive region if the signal it receives from a predecessor node is below a threshold signal strength P_t. Once a node enters this zone, we make at least three consecutive measurements of the signal strength of packets received from the predecessor node, and predict link failure using the Lagrange interpolation.

We store the power strengths of the three signals and their times of occurrence. When two consecutive measurements give the same signal strength, we store the time of the second occurrence. The expected signal strength P of the packets received from the predecessor node is computed as follows:

$$P = \left(\frac{(t-t_1)\times(t-t_2)}{(t_0-t_1)\times(t_0-t_2)} \times P_0 \right) + \left(\frac{(t-t_0)\times(t-t_2)}{(t_1-t_0)\times(t_1-t_2)} \times P_1 \right) + \left(\frac{(t-t_0)\times(t-t_1)}{(t_2-t_0)\times(t_2-t_1)} \times P_2 \right) \quad (\mathbf{7})$$

Where P_0, P_1, P_2 are the measured power strengths at the measurement times t_0, t_1, and t_2, respectively. The time t is the sum of the time needed for discovering an alternative path (Discovery Period), the last measurement time t_2, and the average value of the measurement times t_0, t_1, and t_2. That is:

$$t = t_2 + \left(\frac{t_0 + t_1 + t_2}{3} \right) + \text{Discovery Period} \quad (\mathbf{8})$$

When P is lower than the minimum accepted power (-81 dB) a warning message is sent to the predecessor node. This node then starts a local repair procedure to find alternative paths to the destinations reached using the link to the node that sent the warning message.

5 Conclusion

The Ad hoc networks have recently captured the attention of users because it allows any user to have access to the resources of networks at anytime without regard to location or mobility. Many routing protocols have been used in order to facilitate communication between nodes in a Manets. Routing protocols typically include mechanisms for route discovery and route maintenance. The performance of these protocols depends on the route maintenance mechanism they use. The movement of network nodes can cause routes to become unusable. A node on a route from a source to a destination may become unreachable from its predecessor node because of node movement. Repair mechanisms have some inconveniences. One of those inconveniences is that a maintenance operation starts after link break happens which have a damaging impact on the routing operation. Many studies focuses on anticipate link break using predictive or preemptive techniques.
In this paper, we have present in some detail studies that focuses on predictive and preemptive route maintenance and detailed our new approach that combine both predictive and preemptive route maintenance to gather benefic aspects of previous approaches.

References

1. Belding-Royer, E., and Perkins, C.: Evolution and future directions of the ad hoc on-demand distance-vector routing protocol. Vol. 1 No. 1. Ad Hoc Networks Journal (2003) 125–150.
2. Belding-Royer, E., Perkins, C., and Das, S.: Ad hoc On-Demand Distance Vector (AODV) Routing. draft-ietf-manet-aodv-13.txt. INTERNET DRAFT (2003)
3. Cahill, A., De Leon, P., and Sreenan C.: Link cache extensions for predictive routing and repair in ad hoc wireless networks. in Proceedings of the 4th IEEE Conference on Mobile and Wireless Communication Networks (MWCN) (2002) 43 – 52.
4. Crisòstomo, S., Sargento, S., Brandão, P., and Prior, R.: Improving AODV with Preemptive Local Route Repair: Technical Report Series: DCC-2003-0, Universidade do Porto, Portugal, 2003.
5. Su, W., Lee, S.-J., and Gerla, M.: Mobility Prediction and Routing in Ad hoc Wireless Networks. International Journal of Network Management. Wiley & Sons (2000).
6. Goff, T., Abu-Ghazaleh , N., Phatak, D., and Kahvecioglu, R.: Preemptive routing in ad hoc networks. Journal of Parallel and Distributed Computing (2003) 123–140.
7. Johnson, D., and Maltz, D.: Dynamic Source Routing in Ad Hoc Wireless Networks. Mobile Computing. Kluwer Academic Publishers (1996) 153-181.
8. Srinath, P., Abhilash, P., and Sridhar, I.: Router Handoff: A Preemptive Route Repair Strategy for AODV. Personal Wireless Communications. IEEE International Conference (2002).
9. Boukerche, A., and Zhang, L.: A performance evaluation of a pre-emptive on-demand distance vector routing protocol for mobile ad hoc networks. Vol. 4. wireless communications and mobile computing (2004) 99–108.
10. Tuch, B.: Development of WaveLAN, an ISM Band Wireless LAN. Vol. 4 No. 72. AT&T Technical Journal (1993) 27–33.
11. Boukli Hacene, S., Lehireche, A., Meddahi, A.: Predictive Preemptive Ad Hoc On-Demand Distance Vector Routing. Vol. 19 No. 2. Malaysian Journal of Computer Science (2006)189–195.

Nouvel algorithme de résolution d'un programme quadratique convexe à variables mixtes

S. RADJEF - M.0. BIBI

Laboratoire de Modélisation et d'Optimisation des Systèmes
Université de Béjaia, 06000 (Algérie)

sonia.radjef@caramail.com
mohandbibi@yahoo.fr

Résumé. *Dans ce papier, nous présentons un nouvel algorithme pour résoudre un programme quadratique convexe, où les variables de décision sont de deux types : variables bornées supérieurement et inférieurement et variables nonnégatives. La méthode est une généralisation de la méthode directe du support développée par les professeurs R. Gabasssov et F.M Kirillova. La particularité de cette méthode est le fait qu'elle évite toute transformation préliminaire des variables de décision. Elle manipule les bornes telles qu'elles se présentent initiallement. En outre, elle possède un critère de suboptimalité qui permet d'arrêter l'algorithme avec une précision choisie à l'avance.*
Mots clé *: programme quadratique convexe, variables mixtes, méthode directe du support, suboptimalité,* $\epsilon-$*optimal.*
Abstract. *In this paper, we present a new adaptive algorithm to solve a convex quadratic programming problem, where the decision variables are of two types : upper and lower bounded variables and nonnegative variables. The method is a generalization of the direct support method developed by R. Gabasssov and F.M Kirillova. The particularity of this method is the fact that it avoids the preliminary transformation of the decision variables. It handles the bounds such as they are initially presented, and possesses a suboptimal criterion which stops the algorithm with the desired accuracy.*
Key words *: convex quadratic program, mixed variables, direct support method, suboptimality,* $\epsilon-$*optimal.*

1 Introduction

La méthode la plus classique pour résoudre ce type de problème est la méthode du simplexe quadratique de Wolfe [9].
Ici, nous proposons une nouvelle méthode directe du support basée sur le principe des méthodes adaptées développées par les Professeurs R. Gabassov et F.M. Kirillova [3, 4, 6]. Ces méthodes sont des méthodes intermédiaires entre les méthodes d'activation des contraintes et celles des points intérieures.

La particularité de cette méthode est, entre autres, le fait qu'elle tient compte des spécificités du problème et de ce fait, manipule les contraintes telles qu'elles se présentent initiallement sans avoir à effectuer une transformation préliminaire des variables de décision, cela évite d'agrandir les dimensions du problème, d'où le gain important en temps et espace mémoire qu'elle présente. En outre, cet algorithme est doté d'un critère d'arrêt qui permet d'obtenir une solution ϵ-optimale avec une précision $\epsilon \geq 0$ choisie à l'avance.

Le principe de cette méthode est simple, partant d'une solution réalisable initiale, chaque itération consiste à trouver une direction d'amélioration et un pas maximal le long de cette direction de façon à améliorer la valeur de la fonction objectif tout en veillant à ne pas sortir du domaine réalisable déterminé par les contraintes du problème.

Le problème de la programmation quadratique convexe à variables mixtes se présente sous la forme standard suivante :

$$F(x,y) = \frac{1}{2}x^t D_1 x + c_1^t x + \frac{1}{2}y^t D_2 y + c_2^t y \longrightarrow \min, \tag{1}$$

$$Ax + Hy = b, \tag{2}$$

$$d^- \leq x \leq d^+, \tag{3}$$

$$y \geq 0, \tag{4}$$

où c_1 et x sont des n_x-vecteurs, c_2 et y sont des n_y-vecteurs, A et H des matrices d'ordre $m \times n_x$ et $m \times n_y$ respectivement, avec $rang(A|H) = m < n_x + n_y$, $b \in \mathbb{R}^m$, $d^- \in \mathbb{R}^{n_x}$ et $d^+ \in \mathbb{R}^{n_x}$; les matrices D_1 et D_2 sont symétriques et semi-définies positives.
Soient les ensembles d'indices suivants :
$I = \{1, 2, \cdots, m\}$, $J_x = \{1, 2, \cdots, n_x\}$, $J_y = \{n_x + 1, n_x + 2, \cdots, n_x + n_y\}$.
$J = J_x \cup J_y$, $J_x = J_{x_B} \cup J_{x_N}$, $J_y = J_{y_B} \cup J_{y_N}$ avec $J_{x_B} \bigcap J_{x_N} = \phi$ et $J_{y_B} \bigcap J_{y_N} = \phi$, $|J_{x_B}| + |J_{y_B}| = m$.
Nous posons $J_B = J_{x_B} \cup J_{y_B}$, $J_N = J \setminus J_B = J_{x_N} \cup J_{y_N}$ et notons par A_H la matrice $(A|H)$ d'ordre $m \times (n_x + n_y)$. On peut alors écrire et fractionner les vecteurs de la manière suivante :
$x = x(J_x) = (x_j,\ j \in J_x)$,

$$x = \begin{pmatrix} x_B \\ -- \\ x_N \end{pmatrix}, \quad x_B = x(J_{x_B}) = (x_j,\ j \in J_{x_B}), \quad x_N = x(J_{x_N}) = (x_j,\ j \in J_{x_N});$$

$y = y(J_y) = (y_j,\ j \in J_y)$,

$$y = \begin{pmatrix} y_B \\ -- \\ y_N \end{pmatrix}, \quad y_B = y(J_{y_B}) = (y_j,\ j \in J_{y_B}), \quad y_N = y(J_{y_N}) = (y_j,\ j \in J_{y_N});$$

$A = (a_j, j \in J_x) = (A_B|A_N)$, $A_B = A(I, J_{x_B})$, $A_N = A(I, J_{x_N})$,
$H = (h_j, j \in J_y) = (H_B|H_N)$, $H_B = H(I, J_{y_B})$, $H_N = H(I, J_{y_N})$;
$A_H = A_H(I, J) = (a_{H_j}, j \in J_x \cup J_y) = (A_{H_B}|A_{H_N})$,
$A_{H_B} = A_H(I, J_{x_B} \cup J_{y_B}) = (A_B|H_B)$, $A_{H_N} = A_H(I, J_{x_N} \cup J_{y_N}) = (A_N|H_N)$.

Définitions.

- *Un vecteur (x, y) vérifiant les contraintes (2)-(4) est appelé plan ou solution réalisable du problème (1)-(4).*

- *Un plan* (x^0, y^0) *est dit optimal si*

$$F(x^0,y^0)=\frac{1}{2}{x^0}^t D_1 x^0 + c_1^t x^0 + \frac{1}{2}{y^0}^t D_2 y^0 + c_2^t y^0 = \min(\frac{1}{2}x^t D_1 x + c_1^t x + \frac{1}{2}y^t D_2 y + c_2^t y),$$

où (x, y) *est pris parmi tous les plans du problème (1)-(4).*
- *Un plan* (x^ϵ, y^ϵ) *est appelé* ϵ*-optimal ou suboptimal si*

$$F(x^\epsilon,y^\epsilon)-F(x^0,y^0)\le\epsilon,$$

où ϵ *est un nombre positif ou nul, donné à l'avance et* (x^0, y^0) *est une solution optimale du problème (1)-(4).*
- *L'ensemble* $J_B = J_{x_B} \cup J_{y_B} \subset J,\ |J_B| = m$ *est appelé support des contraintes si*

$$\det A_{H_B} = \det(A(I, J_{x_B}), H(I, J_{y_B})) = \det(A_B, H_B) \neq 0.$$

- *Le couple* $\{(x, y), (J_{x_B}, J_{y_B})\}$*, formé d'une solution réalisable* (x, y) *et d'un support* (J_{x_B}, J_{y_B})*, est appelé plan de support.*
- *Le plan de support est dit non dégénéré, si*

$$d_j^- < x_j < d_j^+, \quad \textit{pour tout} \quad j \in J_{x_B} \qquad \textit{et} \qquad y_j > 0, \quad \textit{pour tout} \quad j \in J_{y_B}.$$

2 Formule d'Accroissement de la Fonction Objectif

Soit $\{(x, y), (J_{x_B}, J_{y_B})\}$ un plan de support du problème (1)-(4) et considérons un autre plan quelconque $(\bar{x}, \bar{y}) = (x + \Delta x, y + \Delta y)$.

On note par :

$D = \begin{pmatrix} D_1(J_x, J_x) & 0 \\ 0 & D_2(J_y, J_y) \end{pmatrix}$: la matrice d'ordre $(n_x + n_y) \times (n_x + n_y)$,

$z = \begin{pmatrix} x \\ y \end{pmatrix}$: le vecteur d'ordre $(n_x + n_y)$,

$c = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}$: le vecteur d'ordre $(n_x + n_y)$.

En posant

$$\Delta z = \begin{pmatrix} \Delta z_B \\ -- \\ \Delta z_N \end{pmatrix},\quad \Delta z_B = \Delta z(J_B) = \begin{pmatrix} \Delta x_B \\ \Delta y_B \end{pmatrix},\quad \Delta z_N = \Delta z(J_N) = \begin{pmatrix} \Delta x_N \\ \Delta y_N \end{pmatrix},$$

$$g(z) = Dz + c = \begin{pmatrix} D_1 x + c_1 \\ D_2 y + c_2 \end{pmatrix} = \begin{pmatrix} g_1(x) \\ g_2(y) \end{pmatrix} = \begin{pmatrix} g_B \\ g_N \end{pmatrix},\quad g_B = g(J_B) = \begin{pmatrix} g_{1_B} \\ g_{2_B} \end{pmatrix},$$

$$g_N = g(J_N) = \begin{pmatrix} g_{1_N} \\ g_{2_N} \end{pmatrix},\quad g_{1_B} = g_1(J_{x_B}),\quad g_{2_B} = g_2(J_{y_B}),\quad g_{1_N} = g_1(J_{x_N}),\quad g_{2_N} = g_2(J_{y_N}).$$

L'accroissement de la fonction objectif s'écrit alors :

$$F(\overline{z}) - F(z) = \left(-g_B^t A_{H_B}^{-1} A_{H_N} + g_N^t\right) \Delta z_N + \frac{1}{2}(\Delta z_N)^t \begin{pmatrix} -A_{H_B}^{-1} A_{H_N} \\ I_N \end{pmatrix}^t D \begin{pmatrix} -A_{H_B}^{-1} A_{H_N} \\ I_N \end{pmatrix} (\Delta z_N)$$

où $I_N = I(J_N, J_N)$ est une matrice identité d'ordre $(n_x + n_y - m)$.

Nous posons

$$Z = \begin{pmatrix} -A_{H_B}^{-1} A_{H_N} \\ I_N \end{pmatrix},\ M = Z^t D Z,$$

et définissons le vecteur des potentiels u ainsi que le vecteur des estimations E comme suit :

$$u^t = g_B^t A_{H_B}^{-1},\ E^t = g^t - u^t A_H = (E_B^t, E_N^t),$$

où

$$E_B^t = 0,\ E_N^t = g_N^t - u^t A_{H_N}.$$

Finalement, l'accroissement de la fonction objectif aura la forme suivante :

$$F(\overline{z}) - F(z) = E_N^t \Delta z_N + \frac{1}{2}(\Delta z_N)^t M(\Delta z_N). \tag{5}$$

Comme

$$g(z) = Dz + c = \begin{pmatrix} D_1 & 0 \\ 0 & D_2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} = \begin{pmatrix} D_1 x + c_1 \\ D_2 y + c_2 \end{pmatrix} = \begin{pmatrix} g_1(x) \\ g_2(y) \end{pmatrix},$$

on peut encore exprimer le vecteur des potentiels u ainsi que le vecteur des estimations E comme suit :

$$u^t = (g_{1_B}^t, g_{2_B}^t) A_{H_B}^{-1}, E^t = (E_B^t, E_N^t),$$

avec

$$E_B^t = (E_{x_B}^t, E_{y_B}^t) = (0,0),\ E_N^t = (E_{x_N}^t, E_{y_N}^t), E_{x_N}^t = g_{1_N}^t - u^t A_N, E_{y_N}^t = g_{2_N}^t - u^t H_N.$$

L'accroissement de la fonction objectif peut encore s'écrire sous la forme suivante :

$$F(\overline{z}) - F(z) = F(\overline{x}, \overline{y}) - F(x,y) = E_{x_N}^t \Delta x_N + E_{y_N}^t \Delta y_N + \frac{1}{2} \begin{pmatrix} \Delta x_N \\ \Delta y_N \end{pmatrix}^t M \begin{pmatrix} \Delta x_N \\ \Delta y_N \end{pmatrix}. \tag{6}$$

Remarque 1. Ayant les matrices $D_1 \geq 0$ et $D_2 \geq 0$, alors $D = \begin{pmatrix} D_1 & 0 \\ 0 & D_2 \end{pmatrix} \geq 0$; donc $M = Z^t D Z \geq 0$.

3 Critère d'optimalité

Théorème 1. *Soit $\{(x,y), (J_{x_B}, J_{y_B})\}$ un plan de support des contraintes du problème (1). Alors les relations suivantes :*

$$\begin{cases} E_{x_j} \geq 0, & si \quad x_j = d_j^-, & j \in J_{x_N}, \\ E_{x_j} \leq 0, & si \quad x_j = d_j^+, & j \in J_{x_N}, \\ E_{x_j} = 0, & si \quad d_j^- < x_j < d_j^+, & j \in J_{x_N}, \\ E_{y_j} \geq 0, & si \quad y_j = 0, & j \in J_{y_N}, \\ E_{y_j} = 0, & si \quad y_j > 0, & j \in J_{y_N}, \end{cases} \tag{7}$$

sont suffisantes pour l'optimalité du plan (x,y).
Ces mêmes relations sont aussi nécessaires, si le plan de support des contraintes est non-dégénéré.

Démonstration. Condition suffisante :

Soit $\{(x,y), (J_{x_B}, J_{y_B})\}$ un plan de support des contraintes du problème (1)-(4) vérifiant les relations (7). Pour tout plan $(\overline{x}, \overline{y})$ du problème (1)-(4), la formule d'accroissement (6) nous permet d'écrire :

$$F(\overline{x}, \overline{y}) - F(x,y) \geq E_{x_N}^t \Delta x_N + E_{y_N}^t \Delta y_N,$$

car la matrice M est semi-définie positive. D'où

$$F(\overline{x}, \overline{y}) - F(x,y) \geq \sum_{j \in J_{x_N}, E_{x_j} > 0} E_{x_j}(\overline{x}_j - x_j) + \sum_{j \in J_{x_N}, E_{x_j} < 0} E_{x_j}(\overline{x}_j - x_j) + \sum_{j \in J_{y_N}, E_{y_j} > 0} E_{y_j}(\overline{y}_j - y_j).$$

Alors, on aura

$$F(\overline{x},\overline{y})-F(x,y) \geq \sum_{j\in J_{x_N},E_{x_j}>0} E_{x_j}(\overline{x}_j-d_j^-)+\sum_{j\in J_{x_N},E_{x_j}<0} E_{x_j}(\overline{x}_j-d_j^+)+\sum_{j\in J_{y_N},E_{y_j}>0} E_{y_j}(\overline{y}_j)\geq 0,$$

où $(\overline{x},\overline{y})$ est une solution réalisable arbitraire du problème (1)-(4).
Par conséquent, le vecteur (x,y) est une solution optimale du problème (1)-(4).
Condition nécessaire.

Soit $\{(x,y),(J_{x_B},J_{y_B})\}$ un plan de support optimal non-dégénéré du problème (1)-(4) et supposons que les relations (7) ne sont pas vérifiées, c-à-d, qu'il existe au moins un indice $j_0\in J_N$, tel que

$$\begin{gathered}
E_{x_{j_0}}>0,\ \text{pour}\ x_{j_0}>d_{j_0}^-,\ j_0\in J_{x_N},\\
\text{ou}\\
E_{x_{j_0}}<0,\ \text{pour}\ x_{j_0}<d_{j_0}^+,\ j_0\in J_{x_N},\\
\text{ou}\\
E_{y_{j_0}}<0,\ \text{pour}\ y_{j_0}=0,\ j_0\in J_{y_N},\\
\text{ou}\\
E_{y_{j_0}}\neq 0,\ \text{pour}\ y_{j_0}>0,\ j_0\in J_{y_N}.
\end{gathered}$$

Nous construisons alors un autre plan $(\overline{x},\overline{y})=(x+\theta l_x, y+\theta l_y)$ où θ est un nombre réel positif, et $l=l(J)=\begin{pmatrix} l_x\\ l_y\end{pmatrix}=\begin{pmatrix} l(J_x)\\ l(J_y)\end{pmatrix}$ est un vecteur de direction que l'on construit comme suit :

Pour ce, deux cas peuvent se présenter

– Si $j_0\in J_{x_N}$, nous posons

$$\begin{cases} l_{x_{j_0}}=-\text{sign}E_{x_{j_0}},\\ l_{x_j}=0, j\neq j_0, j\in J_{x_N},\\ l_{y_j}=0, j\in J_{y_N},\\ l_B=\begin{pmatrix} l_{x_B}\\ l_{y_B}\end{pmatrix}=A_{H_B}^{-1}a_{j_0}\text{sign}E_{x_{j_0}}. \end{cases}$$

– Si $j_0\in J_{y_N}$, nous posons

$$\begin{cases} l_{y_{j_0}}=-\text{sign}E_{y_{j_0}},\\ l_{y_j}=0, j\neq j_0, j\in J_{y_N},\\ l_{x_j}=0, j\in J_{x_N},\\ l_B=\begin{pmatrix} l_{x_B}\\ l_{y_B}\end{pmatrix}=A_{H_B}^{-1}h_{j_0}\text{sign}E_{y_{j_0}}. \end{cases}$$

En vertu de la procédure de construction de la direction d'amélioration l, le vecteur $(\overline{x},\overline{y})$ satisfait la principale contrainte $A\bar{x}+H\overline{y}=b$.

Le vecteur $(\overline{x},\overline{y})$ doit en plus vérifier les inégalités

$$d^-\leq\overline{x}\leq d^+ \qquad et \qquad \overline{y}\geq 0.$$

Soit, en écrivant composante par composante

$$\begin{array}{llll} d_j^- - x_j & \leq \theta l_{x_j} & \leq d_j^+ - x_j, & j\in J_{x_B},\\ d_j^- - x_j & \leq \theta l_{x_j} & \leq d_j^+ - x_j, & j\in J_{x_N}, \end{array} \tag{8}$$

et

$$\theta l_{y_j}\geq -y_j, j\in J_{y_B}, \qquad \theta l_{y_j}\geq -y_j, j\in J_{y_N}. \tag{9}$$

Deux cas peuvent se présenter :

– Si $j_0 \in J_{x_N}$, nous posons :

$$\theta_x = \min\left(\theta_{x_{j_1}}, \theta_{x_{j_0}}\right),$$

où

$$\theta_{x_{j_1}} = \min\left(\theta_{x_j}, j \in J_{x_B}\right), \quad \theta_{x_j} = \begin{cases} \frac{d_j^+ - x_j}{l_{x_j}}, & \text{si} \quad l_{x_j} > 0, \\ \frac{d_j^- - x_j}{l_{x_j}}, & \text{si} \quad l_{x_j} < 0, \\ \infty, & \text{si} \quad l_{x_j} = 0. \end{cases}$$

et

$$\theta_{x_{j_0}} = \begin{cases} d_{j_0}^+ - x_{j_0}, & \text{si} \quad E_{x_{j_0}} < 0, \\ x_{j_0} - d_{j_0}^-, & \text{si} \quad E_{x_{j_0}} > 0. \end{cases}$$

Le nombre θ_y va être calculer comme suit :

$$\theta_y = \min\left(\theta_{y_j}, j \in J_{y_B}\right), \quad \theta_{y_j} = \begin{cases} \frac{-y_j}{l_{y_j}}, & \text{si} \quad l_{y_j} < 0, \\ \infty, & \text{si} \quad l_{y_j} \geq 0. \end{cases}$$

Alors, nous posons

$$\theta^0 = \min\left(\theta_x, \theta_y\right).$$

En vertu de la non-dégénérescence du plan de support $\{(x,y),(J_{x_B}, J_{y_B})\}$, le pas θ^0 construit est strictement positif. Ainsi, le vecteur $\overline{z} = z + \theta l$ est une solution réalisable du problème (1)-(4) pour tout θ tel que $0 < \theta \leq \theta^0$.
La formule d'accroissement de la fonction objectif (6) nous donne alors :

$$\begin{aligned} F(\overline{x},\overline{y}) - F(x,y) =& E_{x_N}^t \Delta x_N + E_{y_N}^t \Delta y_N + \frac{1}{2} \begin{pmatrix} \Delta x_N \\ \Delta y_N \end{pmatrix}^t M \begin{pmatrix} \Delta x_N \\ \Delta y_N \end{pmatrix} \\ =& \theta \left(-|E_{x_{j_0}}| + \frac{1}{2}\theta \begin{pmatrix} l_{x_N} \\ l_{y_N} \end{pmatrix}^t M \begin{pmatrix} l_{x_N} \\ l_{y_N} \end{pmatrix} \right) \\ =& \theta \left(-|E_{x_{j_0}}| + \frac{1}{2}\theta l_N^t M l_N \right). \end{aligned}$$

Pour $\theta > 0$ assez petit, nous aurons alors $F(\overline{x},\overline{y}) - F(x,y) < 0$, ce qui contredit l'optimalité de (x,y).

– si $j_0 \in J_{y_N}$, nous posons

$$\theta_x = \min\left(\theta_{x_j}, j \in J_{x_B}\right), \quad \theta_{x_j} = \begin{cases} \frac{d_j^+ - x_j}{l_{x_j}}, & \text{si} \quad l_{x_j} > 0, \\ \frac{d_j^- - x_j}{l_{x_j}}, & \text{si} \quad l_{x_j} < 0, \\ \infty, & \text{si} \quad l_{x_j} = 0. \end{cases}$$

Le nombre θ_y va être calculer comme suit :

$$\theta_y = \min\left(\theta_{y_{j_1}}, \theta_{y_{j_0}}\right),$$

avec

$$\theta_{y_{j_1}} = \min\left(\theta_{y_j}, j \in J_{y_B}\right), \quad \theta_{y_j} = \begin{cases} \frac{-y_j}{l_{y_j}}, & \text{si} \quad l_{y_j} < 0, \\ \infty, & \text{si} \quad l_{y_j} \geq 0, \end{cases}$$

et

$$\theta_{y_{j_0}} = \begin{cases} y_{j_0}, & \text{si} \quad E_{y_{j_0}} > 0, \\ \infty, & \text{si} \quad E_{y_{j_0}} < 0. \end{cases}$$

Nous définissons

$$\theta^0 = \min\left(\theta_x, \theta_y\right).$$

Pour tout θ tel que $0 < \theta \leq \theta^0$, le vecteur $\overline{z} = z + \theta l$ est une solution réalisable du problème (1)(4). La formule d'accroissement de la fonction objectif (6) nous donne :

$$F(\overline{x}, \overline{y}) - F(x, y) = \theta \left(-|E_{y_{j_0}}| + \frac{1}{2} \theta l_N^t M l_N \right)$$

Pour $\theta > 0$ assez petit, nous aurons encore $F(\overline{x}, \overline{y}) - F(x, y) < 0$, ce qui contredit l'optimalité de (x, y).

Par conséquent, les relations (7) sont suffisantes, et aussi nécessaires pour l'optimalité du plan (x, y), dans le cas où (x, y) est non dégénéré. □

4 Critère de suboptimalité

Pour estimer l'écart qui existe entre la valeur optimale $F(x^0, y^0)$ et une autre valeur $F(x, y)$ d'un plan de support des contraintes quelconque $\{(x, y), (J_{x_B}, J_{y_B})\}$, quand $E_y \geq 0$, nous utilisons la formule suivante :

$$\beta((x, y), (J_{x_B}, J_{y_B})) = \sum_{j \in J_{x_N}, E_{x_j} > 0} E_{x_j}(x_j - d_j^-) + \sum_{j \in J_{x_N}, E_{x_j} < 0} E_{x_j}(x_j - d_j^+) + \sum_{j \in J_{y_N}} E_{y_j} y_j,$$

appelée *estimation de suboptimalité.*

Théorème 2. *(Critère de suboptimalité)*
Soit $\{(x, y), (J_{x_B}, J_{y_B})\}$ un plan de support des contraintes du problème (1) et ϵ un nombre positif ou nul arbitraire.
Si $E_{y_N} \geq 0$ et si $\beta((x, y), (J_{x_B}, J_{y_B})) \leq \epsilon$, alors le plan (x, y) est $\epsilon-$optimal.

Démonstration. Nous avons

$$F(x, y) - F(x^0, y^0) \leq \beta((x, y), (J_{x_B}, J_{y_B})).$$

Si $\beta((x, y), (J_{x_B}, J_{y_B})) \leq \epsilon$ alors, on aura

$$F(x, y) - F(x^0, y^0) \leq \epsilon.$$

Le plan (x, y) est donc $\epsilon-$optimal. Dans le cas particulier où $\epsilon = 0$, la solution réalisable (x, y) est, par conséquent, optimale. □

5 Construction de l'algorithme

Avant de présenter la méthode de résolution, donnons quelques définitions essentielles.

Définitions.

- *On appelle support de la fonction objectif du problème (1), l'ensemble des indices $J_S \subset J_N$ où $J_S = (J_{x_S} \cup J_{y_S})$ et $J_N = (J_{x_N} \cup J_{y_N})$ tel que*

$$detM(J_S, J_S) \neq 0.$$

On posera $J_{x_{NN}} = J_{x_N} \backslash J_{x_S}$, $\quad J_{y_{NN}} = J_{y_N} \backslash J_{y_S}$, $\quad J_{x_{NN}} \cup J_{y_{NN}} = J_{NN}$.

- *L'ensemble des indices $J_P = \{J_B, J_S\}$ est appelé support du problème (1), où $J_B = J_{x_B} \cup J_{y_B}$ est le support des contraintes et $J_S = J_{x_S} \cup J_{y_S}$ est le support de la fonction objectif.*

- *On appelle plan de support du problème (1), la paire $\{(x, y), J_P\}$ formé du plan (x, y) et du support J_P ; il est dit accordé si $E(J_S) = 0$.*

- *La direction $l = \begin{pmatrix} l_x \\ l_y \end{pmatrix}$ où $l_x = l(J_x)$ et $l_y = l(J_y)$, est dite admissible si $Al_x + Hl_y = 0$.*
Elle est dite d'amélioration si en outre $E_x^t l_x + E_y^t l_y < 0$.

Etant donné un nombre réel positif ou nul quelconque ϵ et un plan de support initial $\{(x,y), J_P\}$, le but de l'algorithme est alors de construire un plan $\epsilon-$optimal (x^ϵ, y^ϵ) ou carrément un plan optimal (x^0, y^0).
L'itération de l'algorithme consiste à faire le passage de $\{(x,y), J_P\}$ vers $\left\{(\overline{x},\overline{y}), \overline{J}_P\right\}$ tel que $F(\overline{x},\overline{y}) \leq F(x,y)$. Pour cela, construisons le nouveau plan $(\overline{x},\overline{y})$ de la manière suivante : $(\overline{x},\overline{y}) = (x,y) + \theta(l_x, l_y)$, où $\begin{pmatrix} l_x \\ l_y \end{pmatrix} = l$ est la direction d'amélioration et θ est le pas le long de cette direction.

Dans cet algorithme, on choisira la métrique du simplexe. On ne fera donc varier qu'une seule composante parmi celles qui ne vérifient pas les relations (7).
Pour que l'accroissement de la fonction objectif soit maximal, il faut prendre θ aussi grand que possible et choisir l'indice j_0 tel que :

$$|E_{j_0}| = \max(|E_{x_{j_0}}|, |E_{y_{j_0}}|),$$

avec

$$|E_{x_{j_0}}| = \max\left(|E_{x_j}|,\ j \in J_{x_{NNO}}\right), \qquad |E_{y_{j_0}}| = \max\left(|E_{y_j}|,\ j \in J_{y_{NNO}}\right),$$

où $J_{x_{NNO}}$ et $J_{y_{NNO}}$ sont les sous-ensembles des l'ensemble des indices respectivement J_{x_N} et J_{y_N} qui ne satisfont pas les relations d'optimalité (7).

5.1 Calcul de la direction l

Nous avons deux cas :
– si $|E_{j_0}| = |E_{x_{j_0}}|$, nous posons :

$$\begin{cases} l_{x_{j_0}} = -\text{sign}E_{x_{j_0}}, \\ l_{x_j} = 0, j \neq j_0, j \in J_{x_{NN}}, \\ l_{y_j} = 0, j \in J_{y_{NN}}. \end{cases}$$

Nous calculerons la composante l_S de manière à assurer

$$\overline{E}_j = 0, \qquad j \in J_S.$$

Nous avons :

$$E_N^t = g_N^t - g_B^t A_{H_B}^{-1} A_{H_N} = (g_B^t, g_N^t).\begin{pmatrix} -A_{H_B}^{-1} A_{H_N} \\ I_N \end{pmatrix} = g^t(z)Z.$$

Comme $l = Zl_N$, nous aurons

$$\begin{aligned} \overline{E}_N^t &= g^t(z+\theta l)Z = g^t(z)Z + \theta l^t DZ \\ &= g^t(z)Z + \theta l_N^t Z^t DZ \\ &= E_N^t + \theta l_N^t M. \end{aligned}$$

Finalement, nous avons

$$\overline{E}_N = E_N + \theta M l_N.$$

Comme $E(J_S) = 0$, alors l'équation $\overline{E}(J_S) = 0$ est équivalente à

$$M(J_S, J_S)l(J_S) + M(J_S, J_{NN})l(J_{NN}) = 0.$$

Donc

$$l(J_S) = -M_S^{-1} M(J_S, J_{NN})l(J_{NN}).$$

Puis, nous calculons l_B tel que $A_H l = 0$:

$$\begin{aligned} l_B =& - A_{H_B}^{-1}(A_{H_S} l_S + A_{H_{NN}} l_{NN}) \\ =& - A_{H_B}^{-1}\left(A_H(I, J_S) l(J_S) + A_H(I, j_0) l_{j_0}\right) \end{aligned}$$

En résumé, si $|E_{j_0}| = |E_{x_{j_0}}|$, nous poserons :

$$\left\{ \begin{array}{l} l_{x_{j_0}} = -\text{sign} E_{x_{j_0}}, \\ l_{x_j} = 0, j \neq j_0, j \in J_{x_{NN}}, \\ l_{y_j} = 0, j \in J_{y_{NN}}, \\ l(J_S) = M_S^{-1} M(J_S, j_0) \text{sign} E_{x_{j_0}}, \\ l_B = -A_{H_B}^{-1}\left(A_H(I, J_S) l(J_S) - A(I, j_0) \text{sign} E_{x_{j_0}}\right). \end{array} \right.$$

– Si $|E_{j_0}| = |E_{y_{j_0}}|$, nous poserons :

$$\left\{ \begin{array}{l} l_{y_{j_0}} = -\text{sign} E_{y_{j_0}}, \\ l_{y_j} = 0, j \neq j_0, j \in J_{y_{NN}}, \\ l_{x_j} = 0, j \in J_{x_{NN}}, \\ l(J_S) = M_S^{-1} M(J_S, j_0) \text{sign} E_{y_{j_0}}, \\ l_B = -A_{H_B}^{-1}\left(A_H(I, J_S) l(J_S) - H(I, j_0) \text{sign} E_{y_{j_0}}\right). \end{array} \right.$$

5.2 Calcul du pas θ^0

Nous devons prendre le pas θ^0 comme suit :

$$\theta^0 = \min\left(\theta_x, \theta_y, \theta_F\right).$$

– si $|E_{j_0}| = |E_{x_{j_0}}|$, alors

$$\theta_x = \min(\theta_{x_{j_0}}, \theta_{x_{j_1}}, \theta_{x_{j_S}}),$$

où

$$\theta_{x_{j_0}} = \left\{ \begin{array}{ll} d_{j_0}^+ - x_{j_0}, & \text{si } E_{x_{j_0}} < 0, \\ x_{j_0} - d_{j_0}^-, & \text{si } E_{x_{j_0}} > 0, \end{array} \right.$$

et

$$\theta_{x_{j_1}} = \min(\theta_{x_j}, j \in J_{x_B}), \theta_{x_{j_S}} = \min(\theta_{x_j}, j \in J_{x_S}),$$

où

$$\theta_{x_j} = \left\{ \begin{array}{ll} \frac{d_j^+ - x_j}{l_{x_j}}, & \text{si } l_{x_j} > 0, \\ \frac{d_j^- - x_j}{l_{x_j}}, & \text{si } l_{x_j} < 0, \\ \infty, & \text{si } l_{x_j} = 0. \end{array} \right.$$

Le pas θ_y est égal à

$$\theta_y = \min\left(\theta_{y_{j_1}}, \theta_{y_{j_S}}\right), \theta_{y_{j_1}} = \min\left(\theta_{y_j}, j \in J_{y_B}\right), \theta_{y_{j_S}} = \min\left(\theta_{y_j}, j \in J_{y_S}\right),$$

où

$$\theta_{y_j} = \left\{ \begin{array}{ll} \frac{-y_j}{l_{y_j}}, & \text{si } l_{y_j} < 0, \\ \infty, & \text{si } l_{y_j} \geq 0. \end{array} \right.$$

Quant à θ_F, il se calcule de façon que le passage de (x, y) à $(\overline{x}, \overline{y})$ puisse assurer une diminution maximale de la fonction objectif. Ayant

$$\varphi(\theta) = F(\overline{x}, \overline{y}) - F(x, y) = \theta\left(-|E_{x_{j_0}}| + \frac{1}{2}\theta l_N^t M l_N\right) = \theta\left(-|E_{x_{j_0}}| + \frac{1}{2}\theta\alpha\right),$$

où $\alpha = l_N^t M l_N$.
Nous devons alors avoir :

$$\frac{\partial \varphi(\theta)}{\partial \theta} = 0 \Leftrightarrow -|E_{x_{j_0}}| + \theta\alpha = 0.$$

Nous déduisons :

$$\theta_F = \begin{cases} \frac{|E_{x_{j_0}}|}{\alpha}, & \text{si } \alpha > 0, \\ \infty, & \text{si } \alpha = 0. \end{cases}$$

- Si $|E_{j_0}| = |E_{y_{j_0}}|$ alors

$$\theta_x = \min\left(\theta_{x_{j_1}}, \theta_{x_{j_S}}\right), \theta_{x_{j_1}} = \min\left(\theta_{x_j}, j \in J_{x_B}\right), \theta_{x_{j_S}} = \min\left(\theta_{x_j}, j \in J_{x_S}\right),$$

où

$$\theta_{x_j} = \begin{cases} \frac{d_j^+ - x_j}{l_{x_j}}, & \text{si } l_{x_j} > 0, \\ \frac{d_j^- - x_j}{l_{x_j}}, & \text{si } l_{x_j} < 0, \\ \infty, & \text{si } l_{x_j} = 0, \end{cases}$$

$$\theta_y = \min(\theta_{y_{j_0}}, \theta_{y_{j_1}}, \theta_{y_S}),$$

où

$$\theta_{y_{j_0}} = \begin{cases} y_{j_0}, & \text{si } E_{y_{j_0}} > 0, \\ \infty, & \text{si } E_{y_{j_0}} < 0, \end{cases}$$

et

$$\theta_{y_{j_1}} = \min(\theta_{y_j}, j \in J_{y_B}), \theta_{y_S} = \min(\theta_{y_j}, j \in J_{y_S}),$$

où

$$\theta_{y_j} = \begin{cases} \frac{-y_j}{l_{y_j}}, & \text{si } l_{y_j} < 0, \\ \infty, & \text{si } l_{y_j} \geq 0. \end{cases}$$

Pour θ_F, nous déduisons :

$$\theta_F = \begin{cases} \frac{|E_{y_{j_0}}|}{\alpha}, & \text{si } \alpha > 0, \\ \infty, & \text{si } \alpha = 0. \end{cases}$$

Le nouveau plan est $(\overline{x}, \overline{y}) = (x + \theta^0 l_x, y + \theta^0 l_y)$.

5.3 Changement de support

Si $\beta((\overline{x}, \overline{y}), (J_{x_B}, J_{y_B})) \leq \epsilon$, alors le plan $(\overline{x}, \overline{y})$ est $\epsilon-$optimal et nous pouvons arrêter l'algorithme ; sinon, nous allons changer J_P comme suit :
Si $\theta^0 = \theta_{x_{j_0}} \vee \theta_{y_{j_0}}$, alors $\overline{J}_B = J_B$, $\overline{J}_S = J_S$, $\overline{J}_P = J_P$.
Si $\theta^0 = \theta_{x_{j_1}} \vee \theta_{y_{j_1}}$, alors $\overline{J}_B = (J_B \backslash j_1) \cup j_0$, $\overline{J}_S = J_S$, $\overline{J}_P = \{\overline{J}_B, \overline{J}_S\}$.
Si $\theta^0 = \theta_{x_{j_S}} \vee \theta_{y_{j_S}}$, alors $\overline{J}_B = J_B$ et $\overline{J}_S = J_S \backslash j_S$, $\overline{J}_P = \{\overline{J}_B, \overline{J}_S\}$.
Si $\theta^0 = \theta_F$, alors $\overline{J}_B = J_B$ et $\overline{J}_S = J_S \cup j_S$, $\overline{J}_P = \{\overline{J}_B, \overline{J}_S\}$.
Nous recommencerons alors une nouvelle itération avec le nouveau plan de support $\{(\overline{x}, \overline{y}), \overline{J}_P)\}$, $\overline{J}_P = \{\overline{J}_B, \overline{J}_S\}$ où le support $\overline{J}_P$ satisfait les conditions algébriques

$$det\overline{A}_{H_B} = detA_{H_B}(I, \overline{J}_B) \neq 0 \quad \text{et} \quad det\overline{M}_S = detM(\overline{J}_S, \overline{J}_S) \neq 0.$$

6 Conclusion

La méthode que nous avons proposé est une généralisation de la méthode directe du support développée par les professeurs R. Gabassov et F.M Kirillova. Sa particularité de manipuler les variables de décision telles qu'elles se présentent initiallement est très significatif dans les processus d'analyse post-optimalité et dans la caractérisation de la propriété bang-bang d'une commande optimale durant l'optimisation des systèmes dynamiques quadratiques. Le fait que l'algorithme est doté d'un critère d'arrêt qui peut donner une solution approchée avec une précision choisie à l'avance constitue un intérêt considérable.

Références

[1] M.O. Bibi. *Methods for solving linear-quadratic problems of optimal control.* PhD thesis, University of Minsk, 1985.

[2] M.O. Bibi. Support method for solving a linear-quadratic problem with polyhedral constraints on control. *Optimization*, 37 :139–147, 1996.

[3] R. Gabasov et al. *Constructive methods of optimization.* P.I.-University Press, Minsk, 1984.

[4] R. Gabasov, F.M. Kirillova, O.I. Kostyuhova, and V.M. Rakestsky. *Constructive methods of optimization. P.IV : Convex problems.* University Press, Minsk, 1987.

[5] R. Gabassov. *Adaptive method of linear programming.* Preprints of the university of Karlsruhe, Germany, 1993.

[6] R. Gabassov et al. Solution of linear quadratic extremal problems. *Soviet Math. Dokl.*, 31 :99–103, 1985.

[7] E.A. Kostina and O.I. Kostyuhova. An algorithm for solving quadratic programming problems with linearequality and inequality constraints. *Computational mathematics and mathematical physics*, 41(7) :960–973, 2001.

[8] E.A. Kostina and O.I. Kostyuhova. *A primal-dual active set method for convex quadratic programming problems.* Preprints of the university of Karlsruhe, Germany, 2003.

[9] P. Wolfe. The simplex method for quadratic programming. *Econometrica*, 27 :382–398, 1959.

Optimisation d'un système d'inférence flou par la recherche taboue : Application à la navigation d'un robot mobile

F.Boufera[1], F.Debbat[1], & F.Khelfi[2]
[1] Institut d'Informatique, Centre Universitaire Mustapha Stambouli de Mascara,
Tél. : 079 1165 43, Fax : 045 81 39 98
[2]Département Informatique, Faculté des Sciences, Université d'Oran Es-Sénia
boufera2007@yahoo.fr, debbat_fati@yahoo.fr mf_khelfi@yahoo.fr

Résumé. La navigation autonome d'un robot mobile se résume généralement par un déplacement d'une position initiale vers une destination finale, tout en respectant les contraintes de l'environnement et du robot et sans intervention humaine. L'accomplissement d'une t'elle tâche se fait par des manœuvres non évidentes. De se fait, la programmation d'une stratégie de navigation autonome passe par l'emploi d'une commande intelligente employant des connaissances précises d'un expert. Ce travail propose l'application de la recherche taboue pour l'optimisation d'un système d'inférence flou qui assure la navigation autonome d'un robot mobile dans un environnement inconnu. L'optimisation consiste à trouver la meilleure table de règles d'inférence qui permet au robot une bonne navigation sans collision avec les obstacles.

Mots Clés: Robotique mobile, Navigation, Optimisation, Logique floue, recherche Taboue.

1 Introduction

La robotique est un vaste domaine pluridisciplinaire qui touche plusieurs secteurs industriels et publics tels que les systèmes de surveillance, systèmes de contrôle et AGVs (Véhicules guidées autonomes) [1], [2]. Depuis plusieurs décennies, l'attention de très nombreux chercheurs se focalise vers le développement et la conception des architectures de contrôle d'un robot mobile autonome, c'est-à-dire l'organisation de l'ensemble des moyens logiciels et matériels qui permet de donner vie à un robot et lui confère des capacités d'autonomie et d'adaptation.
Cet article présente deux outils : la logique floue et la recherche taboue pour concevoir et implémenter l'architecture de contrôle pour une navigation autonome d'un robot mobile, en tenant compte de la nature changeante de l'environnement.
La nature de notre problématique nous a suggéré alors, d'utiliser :
- La logique floue comme une technique de contrôle.
- La recherche taboue comme outil ou méthode d'optimisation pour l'apprentissage et l'optimisation du contrôleur flou.
Nous avons opté pour une hybridation entre ces deux approches pour plusieurs raisons :

Le principe de réglage et de commande par la logique floue s'approche de la démarche humaine dans le sens où les variables traitées ne sont pas des variables logiques (au sens de la logique binaire par exemples) mais des variables linguistiques, proches du langage humains. De plus, ces variables linguistiques sont traitées à l'aide de règles qui font référence à une certaine connaissance du comportement du système à commander [3].

La recherche taboue [4] qui est une métaheuristique fondée sur le principe de la recherche locale. Ce principe consiste à explorer l'espace de recherche composé de toutes les solutions réalisables dans le but d'aboutir à la solution optimale.

Dans la section suivante, nous présentons le principe du raisonnement et du contrôle flou. La troisième et la quatrième section présentent l'optimisation par la recherche taboue du système d'inférence pour une navigation autonome, la cinquième illustre les résultats de simulation. Le dernier paragraphe conclu le présent papier.

2 Principe de la navigation par la logique floue

La plate forme choisie pour notre robot mobile est la configuration différentielle comportant deux roues motrices commandées indépendamment, deux roues folles d'équilibrage sont ajoutées à l'avant du robot pour assurer sa stabilité et trois capteurs infrarouges. L'un de ces capteurs est frontal et les deux autres latéraux pour assurer le système de perception du robot mobile. Cette plate forme est très simple à commander, puisqu'il suffit de spécifier les vitesses des deux roues, et permet en plus au robot de tourner sur place.

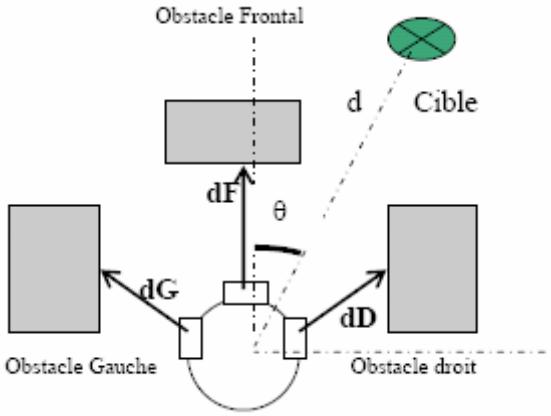


Fig.1. configuration du robot mobile

Le robot réalise un comportement réactif, il ne dispose pas de représentation ou de carte de l'environnement, ni de leur structure interne, il connaît seulement la position de la cible. Le robot doit être capable de modifier son contrôleur flou en cas de changement dans l'environnement.

L'action réflexe du robot est déduite de l'analyse de données suivant les trois cotés du robot (dG, dF, dD : les distances par rapport aux obstacles) et les coordonnées

polaires du point d'arrivée dans le repère du robot (l'orientation de la cible notée γ, et la distance par rapport à la cible notée d).

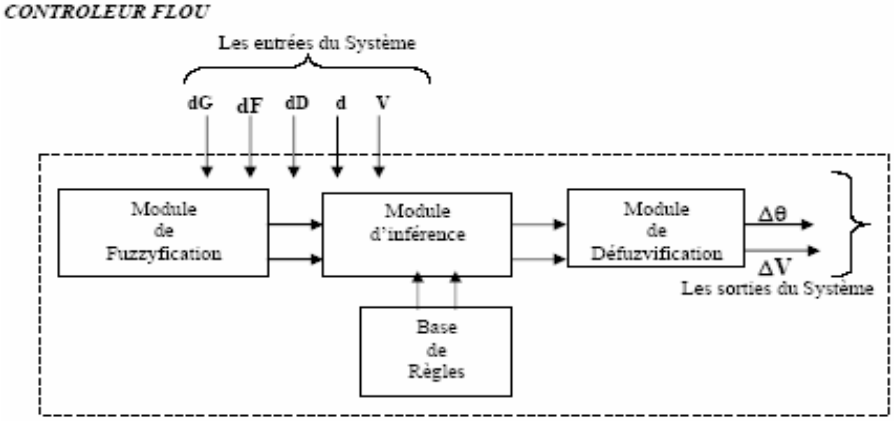


Fig.2. Description du contrôleur flou

Fuzzyfication des données d'entrée

Il s'agit de la fuzzyfication :

- Des distances dG, DF, dD et d sont évaluées par rapport aux deux sous-ensembles flous P et L, qui correspondent respectivement à Prés et Loin.

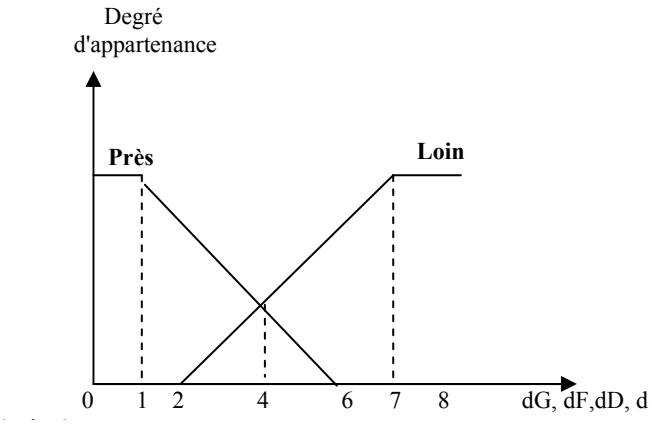


Fig. 3. Représentation des sous ensembles flous de la distance

- De l'angle d'orientation de la cible par rapport au robot γ est représenté par cinq intervalles flous : GG (gauche grand), GP (gauche petit), Z (zéro), DP (droit petit), DG (droit grand) couvrant le demi espace frontal du robot.

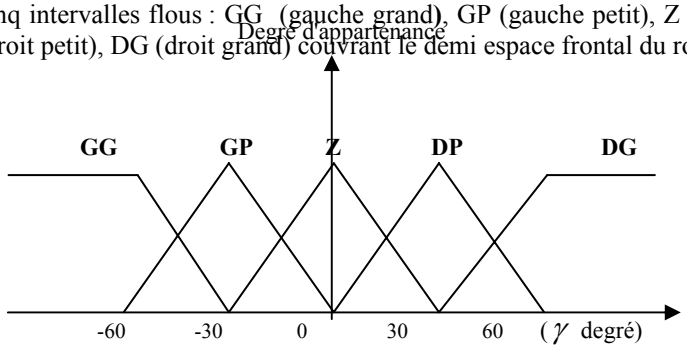


Fig.4. Représentation des sous-ensembles flous de l'entrée γ

Fuzzyfication des sorties

Nous avons deux variables de sortie **:** $\Delta\theta$ et ΔV

- La variable de sortie $\Delta\theta$ (l'angle d'orientation du robot) est représentée par les cinq sous-ensembles flous : TGG (thêta gauche grand), TGP (thêta gauche petit), TZ (thêta zéro), TDP (thêta droite petit), TDG (thêta droite grand).

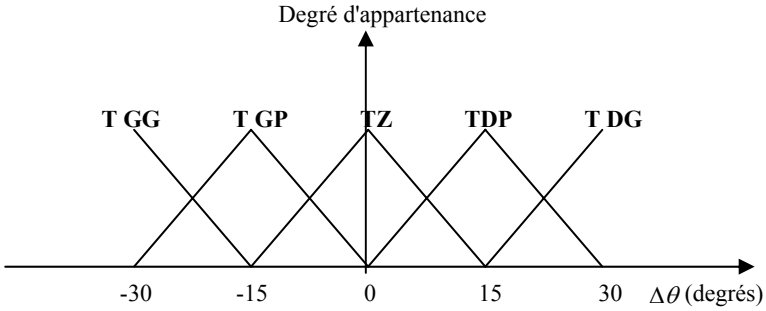


Fig.5. Représentation des sous-ensembles flous de la sortie $\Delta\theta$

- La variation de la vitesse ΔV est décrite par trois sous-ensembles : Dacc (diminuer la vitesses), Zacc (pas de changement de vitesse) et Acce (augmenter la vitesse).

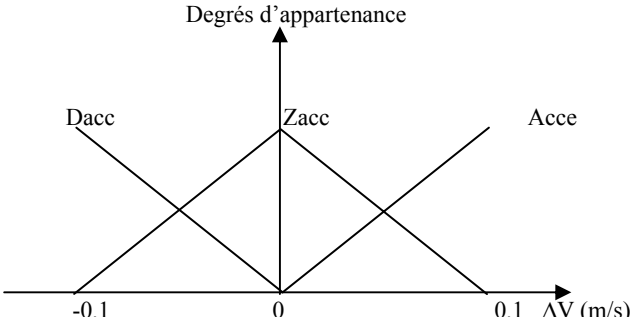


Fig.6. Représentation des sous-ensembles flous de la sortie ΔV

Mécanisme d'inférence

Cette étape concerne l'élaboration des règles, pour définir le comportement attendu du robot selon ses paramètres intrinsèques : les données sur les obstacles, données sur la cible et les consignes de mouvement du robot. Pour chacune des combinaisons des valeurs des variables d'entrées, une action sur les variables de sortie lui est associée. En totalité on obtient 80 règles floues.
Le comportement global du robot durant sa navigation est cerné dans deux procédures primordiales :
L'évitement d'obstacles : Le robot fait appel à cette procédure quand il est confronté à un obstacle.
Rapprochement du but : Il n'y a pas d'obstacles.

Défuzzyfication La sortie du contrôleur est obtenue par la méthode de défuzzyfication dite "moyenne pondéré". Cette étape permet de transformer les valeurs de commande du domaine flou vers le domaine réel (variables physiques). Ce choix est généralement conditionné par un compromis entre la facilité d'implémentation et performance de calcul [2].

3 Principe d'optimisation par la recherche taboue

3.1 Principe général de la recherche taboue

Le modèle proposé dans cet article pour l'apprentissage est basé sur la méthode de recherche tabou [5] qui est une méta heuristique fondée sur le principe de la recherche locale. Ce principe consiste à explorer l'espace de recherche composé de toutes les solutions réalisables dans le but d'aboutir à la solution optimale.

Commençant à partir d'une solution initiale réalisable, le processus consiste, à chaque itération, à choisir la meilleure solution dans le voisinage de la solution courante, même si cette solution n'entraîne pas une amélioration. Ce voisinage est constitué de toutes les solutions obtenues en effectuant une seule *transition* sur la solution courante. Ces solutions sont alors appelées *voisins* de la solution courante.

Afin d'éviter le piège des optima locaux dans lequel ce processus peut être facilement attrapé, la recherche tabou utilise une structure de mémorisation temporaire dans laquelle elle sauvegarde les dernières solutions visitées : la *liste tabou*. En effet, une solution reste interdite pendant un nombre d'itérations égal à la taille de cette liste. Par suite, le meilleur voisin non tabou sera choisi pour la prochaine itération.

3.2 Optimisation du contrôleur flou par la Recherche taboue

Notre travail consiste à déterminer les paramètres d'inférence optimale d'un contrôleur flou, qui donne un comportement idéal pour le robot mobile. Cette tâche est effectuée par un apprentissage automatique avec la recherche taboue.

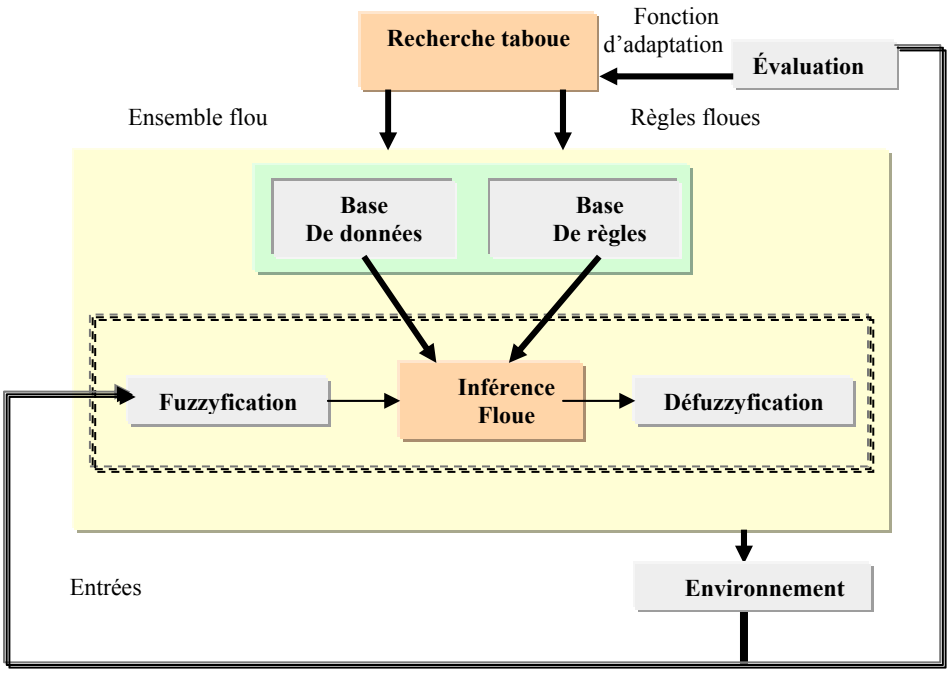


Fig.7. Structure du système d'apprentissage

Optimisation des règles d'inférence

Il existe deux approches d'optimisation :

- Ajustement des parties conclusions des règles d'inférence tout en laissant fixe les parties conditions et c'est l'approche que nous avons adoptée ;
- Ajustement des parties conditions des règles d'inférences tout en laissant fixe les parties conclusions.

Donc nous allons utiliser la recherche taboue pour la recherche de l'ensemble des règles floues qui permet un meilleur comportement du robot (atteindre la cible dans les meilleures conditions, pas de collisions, trajectoire optimale).

L'évaluation de l'ensemble des solutions (tables règles) est effectue sur la base d'une fonction d'adaptation.

Une solution se représente sous la forme d'un vecteur dont le nombre d'éléments correspond au nombre de règles d'inférence (80 règles dans notre cas), et chaque élément représente une conclusion K_i.

Nous avons 15 conclusions possible $K \in \{0, 1, 2, \ldots, 14\}$.

***K* :** correspond à deux variables de sortie (orientation, accélération) (**15** combinaisons possibles = **3** pour la vitesse **x 5** pour l'orientation).

Δθ	TGG	TGP	TZ	TDP	TDG	TGG	TGP	TZ	TDP	TDG	TGG	TGP	TZ	TDP	TDG
ΔV	Dece	Dece	Dece	Dece	Dece	Zacc	Zacc	Zacc	Zacc	Zacc	Acc	Acc	Acc	Acc	Acc
Code	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

Tableau .1 Codification des sorties floues

Génération de la solution initiale :
La solution initiale est un vecteur de 80 éléments (le nombre de règles d'inférence), et dont chaque élément est une combinaison des cas possibles de $\Delta\theta$ et ΔV. Le tableau suivant présente une codification des différents cas possibles (15 cas) pour ces deux paramètres de sortie.

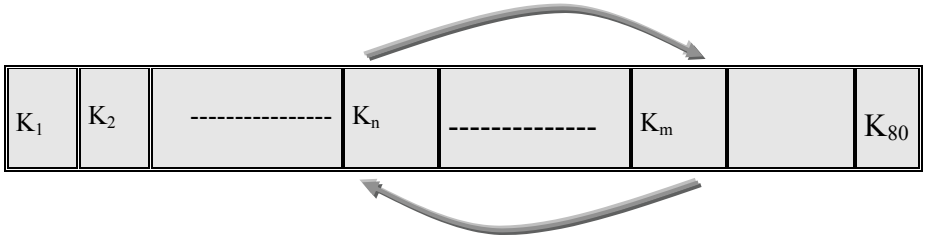
Génération de voisinage :
Chaque solution se présente sous la forme d'un vecteur de 80 éléments :

K_1	K_2	----------	K_{80}

Chaque Ki représente une conclusion d'une règle floue.

Pour générer un voisinage de la solution courante, nous avons appliqué la transformation suivante :

Nous allons prendre 2 éléments du vecteur de la solution courante aléatoirement (2 conclusions). Ensuite nous permutons leurs valeurs.



$$n, m \in [1,80]$$

Nous obtiendrons une nouvelle solution voisine. Nous itérons ce processus pour un certain nombre de voisins qu'il faut fixer au préalable.

La fonction d'adaptation

Pour évaluer les performances de navigation d'un robot mobile, on doit définir les critères essentiels qui fait qu'une navigation est meilleurs qu'autre.

Ces critères sont:

- Le nombre de collision que fait le robot avec les obstacles durant la navigation (**N_collis**);
- La longueur de la trajectoire (**Dist**) ;
- Le temps nécessaire pour atteindre la cible (**Temps**);
- Le nombre de tour (**N_Tour**) effectués par le robot mobile en contournant les obstacles ;

Chaque critère d'évaluation est modélise par une fonction d'adaptation.

$$f_{N_collis} = \begin{cases} \dfrac{1}{\mathbf{N_collis}} & \text{fonction d'état infaisable} \quad (\boldsymbol{f}_{N_collis} \in \{0,1\}) \\ \mathbf{1} & \text{(Sans collision)} \end{cases} \tag{1}$$

$$f_{Dist} = \frac{1}{\mathbf{Dist}} \quad \text{fonction de la longueur du trajet} \quad (\boldsymbol{f}_{Dist} \in \{0,1\}) \tag{2}$$

$$\boldsymbol{f}_{N_tour} = \begin{cases} \dfrac{1}{\mathbf{N\ tour}} & \text{fonction des changements d'orientation} \quad (\boldsymbol{f}_{N_tour} \in \{0,1\}) \\ \mathbf{0} & N_tour \geq N_tour_max \end{cases} \tag{3}$$

La fonction d'adaptation du comportement du robot est la combinaison des trois fonctions :

$$f_{traj} = \frac{f_{N_collis}\left(af_{Dist} + bf_{N_tour}\right)100}{a+b} \tag{4}$$

Avec **a, b** des coefficients de pondération.

La tentative est de pondérer chaque partie de la fonction $\boldsymbol{f}$ traj, s'accordant à leur importance dans l'objectif de l'algorithme. La partie qui a une grande influence sur le trajet est la fonction d'état infaisable $\boldsymbol{f}_{N_collis}$.

- **Pénalisation des mauvais comportements**

Nous pouvons arrêté un comportement mauvais si :

- ❖ Il dépasse le temps max de navigation,
- ❖ Il dépasse le nombre max de collision,
- ❖ Il dépasse le nombre max de Tour.

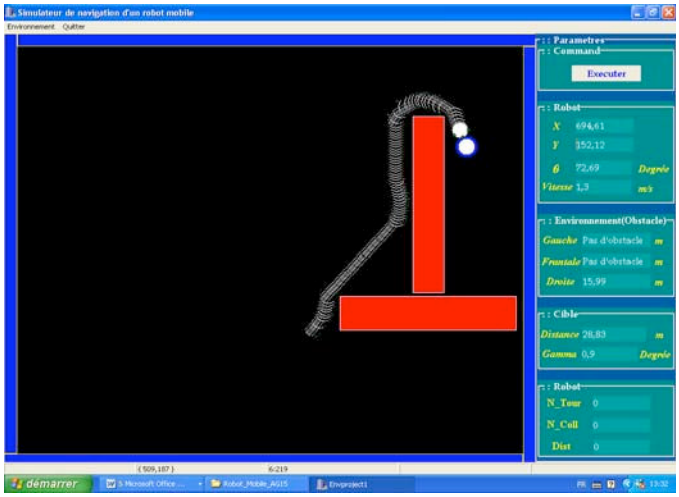
La pénalisation sur la fonction d'adaptation sera faite comme suite :

Si temps ≥ temps_max **Alors** $f$traj = μ * $f$traj
Si N_collisions ≥ N_collisions_max **Alors** $f$traj = μ * $f$traj ;
Si Dist tend vers ***0*** **et** temps ≥ temps_max **Alors** $f$traj = ***0.***

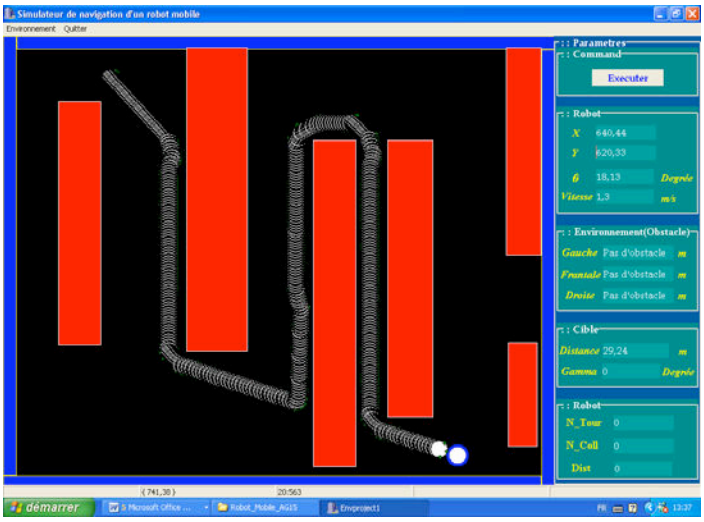
Avec μ et le coefficient de pénalisation qui est égale à ***0.1***.

4 Simulation et Résultats

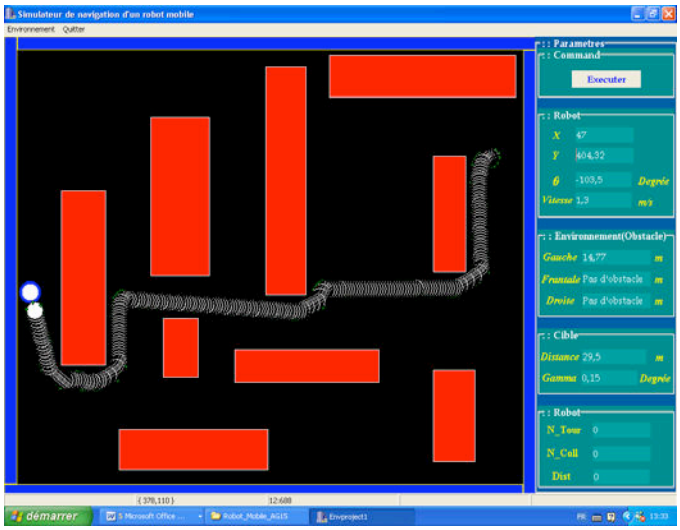
Plusieurs tests de navigation ont été simulés pour différentes configurations de la cible et du robot et sur plusieurs environnements différents. Les figures suivantes montrent le chemin pris par le robot pour atteindre sa cible.



(a) environnement 1



(b) environnement 2



(c) environnement 3

Fig.8. Le robot atteint toujours sa cible en parcourant le chemin le plus court Pendant la navigation (environnement 1,2 et 3).

D'après ces figures, nous remarquons bien les performances et l'efficacité des approches adoptées (RT et la logique floue) dans la navigation du robot mobile dans des environnements complexes et changeants. Le robot arrive toujours à atteindre sa cible sans collision avec les obstacles ni avec les parois de l'environnement tout en choisissant le chemin le plus court.

5 Conclusion générale

Le problème de navigation autonome d'un robot mobile, est un des plus importants et récents problèmes de la robotique mobile. Plusieurs recherches ont été faites pour résoudre ce problème, certaines sont basées sur des approches classiques d'autres sur des approches avancées. Dans cet article, nous avons présenté une solution à ce problème par le développement d'une architecture de contrôle basé sur la logique floue et la recherche taboue. Les résultats de simulation obtenus sont satisfaisants du fait que le robot arrive à atteindre sa cible malgré la complexité de l'environnement où il se trouve. Ceci ouvre la voix vers l'expérimentation de ces résultats sur un robot mobile réel.

Références bibliographiques

1. Trahanias. P.E et al.(). « Navigational support for robotic wheel chair platforms :an approach that combines vision and range sensors ». Proceeding of the .IEEE international conference on robotics and automation, 1997.
2. Wijesoma. W.S , Khaw.P.P and Teak. E.K. « Sensor modelling and fusion for fuzzy navigation of an AGV ».International journal of robotic and automation, 2001.
3. Buhler. H. « Réglage par la logique floue ».Presses polytechniques et universitaires Romandes, 1994.
4. GLOVER (F.), A user's guide to tabu search. Annals of Operations Research, **41**, n° 3, pp. 3-28, 1993.
5. PIERRE (S.), QUINTÉRO (A.),MONTGOLFIER (A.), Approches heuristiques pour l'affectation de cellules aux commutateurs dans les réseaux mobile. IEEE Canadian Review, 2004.

Vers une représentation formelle des schémas semi-structurés

Amar ZERDAZI[1], Myriam LAMOLLE[1], Ludovic MENET[1, 2]

[1] LINC – Université Paris VIII
140, rue de la Nouvelle France 93000 - Montreuil, France
{a.zerdazi, m.lamolle, l.menet}@iut.univ-paris8.fr
[2] Orchestranetworks
75, Boulevard Haussman 75008 - Paris, France
ludovic.menet@orchestranetworks.com

Abstract. Les données semi-structurées sont devenues de plus en plus importantes avec la naissance du langage XML et l'évolution des technologies associées. Ainsi, le passage des DTDs vers XML-Schema a-t-il permis une amélioration dans la représentation structurelle de ces données par une meilleure définition des types de données, de leurs cardinalités, de leurs réutilisations, etc. Cependant, dans le cas des sources de données hétérogènes, les schémas XML seuls, n'offrent pas assez de sémantique pour pouvoir intégrer automatiquement ces sources hétérogènes. Dans cet article, nous proposons un modèle de données sémantiques pour les données semi-structurées appelé CRP (Concept-Relation-Propriété). Ce modèle a pour but principal de pouvoir unifier des représentations issues de modélisations telles que, OEM, DTD, XSD, DSD, etc. CRP ne reflète pas seulement la structure arborescente des données semi-structurées, mais il fait la distinction entre concepts, relations et propriétés. Il est également possible de spécifier par exemple le degré et la catégorie des relations et indiquer si une propriété appartient à une relation ou à un concept. De telles informations sont manquantes dans les modèles semi-structurés existants, et qui sont essentielles pour la non-redondance, l'optimisation, le stockage des données semi-structurées.

Keywords : données semi-structurées, intégration de données, schémas XML.

1 Introduction

Souvent les données du Web sont des données semi-structurées caractérisées par des structures irrégulières, dynamiques ou inconnues. L'intérêt des données semi-structurées dont le standard dominant est le format XML [15] provient de la facilité à permettre l'échange et l'intégration des données entre applications. Cette nouvelle représentation des données a conduit à définir des *modèles* et des systèmes plus adaptés. Ainsi malgré la structure intrinsèque d'un document XML, une structure commune plus générale peut être toutefois associée à un ensemble de documents. Cette

structure énoncée par des DTDs [3] *(Document Type Definition)* est remplacée progressivement par des XSDs *(XML-Schema Definition)* [16] qui visent à pallier la pauvreté sémantique de ces premières. Notamment, XML-Schema permet d'assurer l'intégrité des données (typage de données, cardinalité, etc.), de réutiliser des structures de données (héritage, surcharge, etc.) et donc d'assurer la qualité des données échangées. Cette évolution diminue le fossé entre des représentations relationnelles, des représentations « objets » avec des représentations semi-structurées. Cependant, les modèles de données existants [4], [5], [9] ne permettent pas la définition d'un modèle commun pour les schémas des bases de données traditionnelles et les schémas de données semi-structurées. Or, il est indispensable de permettre l'échange et la coopération entre des données provenant de sources classiques (hiérarchiques, relationnelles, objets) et des sources semi-structurées (data guide, OEM, etc.). Les techniques de *médiation* ont dû être adaptées pour prendre en compte ces nouveaux types de données.

Dans cet article, nous proposons un modèle de données que nous appelons *CRP (Concept-Relation-Propriété)* qui permet de représenter les données semi-structurées, tout en capturant davantage la sémantique portée par ces données. Les données semi-structurées sont souvent stockées dans des fichiers à plat ou des entrepôts, telles que des bases de données relationnelles [13], des bases de données orienté-objet [11] ou dans des systèmes semi-structurés spécialisés, tel que Strudel [7], Lore [12], Lotus Notes [10] et Tamino [14]. Comme dans les SGBDs traditionnels, le stockage et l'accès efficace, la minimisation des mises à jour sont importante et ne peuvent pas être réalisés sans connaissance de la sémantique des données. Un des avantages majeurs des données semi-structurées et en particulier XML, c'est que le stockage des données est complètement indépendant des détails de la présentation. Cette dernière est définie par des feuilles de styles, qui sont similaires à la définition de vues dans les applications des bases de données traditionnelles. Afin de définir des présentations significatives avec des mises à jour simples, le concepteur doit connaître forcément la sémantique des ses données.

Dans notre approche, nous nous intéressons à la structure de ces données, autrement dit aux schémas de données et non pas aux données elles-mêmes, du fait que la taille du schéma est beaucoup moins importante que le volume des données. Aussi, le contenu des données ne garantit pas toujours une représentation complète et disponible.

Nous présentons le reste de l'article comme suit. La section 2 présente quelques modèles de représentation des données semi-structurées. La section 3 illustre une description détaillée du modèle *CRP*. La section 4 concerne le passage d'un schéma semi-structuré vers le modèle *CRP*. Pour cela, nous avons choisi comme cas pratique les schémas XML de type XSD. Enfin, une conclusion et nos futurs travaux font l'objet de la section 5.

2. Représentation des données semi-structurées

Les premiers systèmes d'intégration de données ont utilisé les modèles relationnels ou objets comme modèles communs. Cependant, l'intégration de sources de données peu

ou non structurées est difficile avec ces modèles dédiés à représenter spécifiquement des bases de données. Les modèles de données semi-structurées ont été conçus pour représenter facilement des données irrégulières provenant de sources hétérogènes, structurées ou non [2]. Ce sont des modèles flexibles qui permettent de représenter des données irrégulières en mélangeant la structure et les données [1].

2.1 OEM

Le modèle OEM *(Object Exchange Model)* [12] est l'un des modèles proposés pour les données semi-structurées. Il a été conçu à l'origine pour le projet TSIMMIS [8] pour assurer l'intégration de données de différentes sources. Les données en OEM sont sans schéma et autodescriptives ; nous pourrions les percevoir comme un graphe dirigé et étiqueté dont les noeuds sont des objets (comme illustré à la figure 1(a)). Un objet OEM peut être perçu comme un quadruplet *(oid, étiquette, type, valeur)*. Un objet[1] OEM est constitué d'un identificateur d'objet unique *(par exemple &8)*, d'une étiquette descriptive *(adresse)*, d'un type *(chaîne de caractères)* et d'une valeur *("Paris")*. Les objets sont soit atomiques, soit complexes. Un *objet atomique* contient une valeur d'un type de base (par exemple entier ou chaîne de caractères) et se reconnaît dans le diagramme au fait qu'il n'a pas de flèche sortante. Tous les autres objets sont qualifiés d'*objets complexes*, dont le type est défini par un ensemble d'identificateurs de type, et sont reconnaissables dans le diagramme car ils comportent une ou plusieurs flèches sortantes. Un objet complexe peut être le parent d'un nombre quelconque d'*objets enfants* OEM et un objet OEM peut avoir plusieurs *objets parents*, ce qui permet de construire des réseaux de complexité arbitraire, de manière à modéliser les associations parmi les données. Le modèle OEM a été conçu spécialement pour gérer l'aspect incomplet des données, ainsi que l'irrégularité de leur structure et de leur type.

2.2 Les guides de données

La connaissance de la structure de la base de données est importante dans la formulation de requêtes significatives. De même, le processeur de requête exige une certaine connaissance de la structure de la base de données pour traiter efficacement une requête. Malheureusement, les données semi-structurées peuvent ne pas avoir de schéma et que le schéma doit alors être découvert à partir des données. Lorel [12] offre une caractéristique innovante sous la forme d'un *guide de données (DataGuide*, *en anglais)* un résumé structurel de la base de données, généré et entretenu dynamiquement, qui sert de schéma dynamique [9]. Un guide de données possède trois propriétés : *(i) la concision,* tout chemin d'étiquette dans la base de données n'apparaît qu'une et une seule fois dans le guide de données. *(ii) la précision,* tout chemin d'étiquette dans le guide de données existe dans la base de données initiale. *(iii) le confort.* Le guide de données est un objet OEM (ou XML), de sorte qu'il peut être

[1] Un objet OEM ne possède pas de comportement. En effet, il n'est pas possible de définir de méthodes sur un objet OEM.

stocké et nous pouvons y accéder à l'aide des mêmes techniques que dans la base de données source.

La figure 2(b) présente un guide de données correspondant aux données de la figure 1(a).

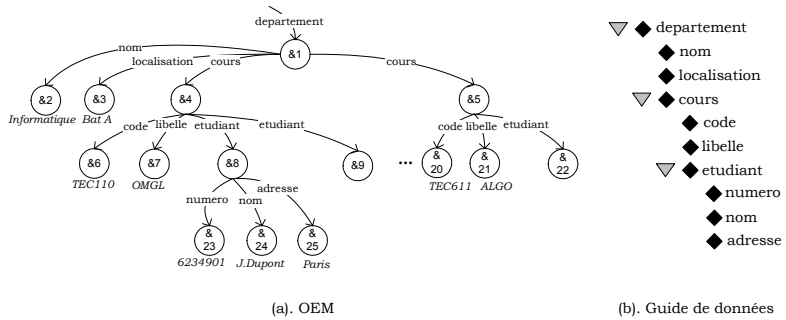


Fig. 1. Représentation d'une structure de données par OEM et guide de données.

2.3 XML/XML-Schema

Le langage XML *(eXtensible Markup Language)* connaît depuis son apparition en 1996 un succès considérable. Recommandation W3C en 1998, XML est devenu un standard pour les documents structurés. Sans en exposer tous les détails, faisons-en une rapide présentation. XML est un sous-ensemble de SGML. Un document XML est donc structuré en éléments. Les balises marquent le début et la fin de chaque élément. Les éléments peuvent contenir du texte et éventuellement d'autres éléments. L'ensemble des données du document XML est contenu dans un élément unique appelé racine, élément qui contient tous les autres éléments.

XML-Schema [16] est uns standard permettant de définir et de typer un document XML dans un formalisme proche de celui des bases de données. Il devient alors possible de définir des schémas des bases de données comme dans un modèle objet. En particulier, les schémas XSD présentent les avantages suivants.

Syntaxe XML : Adopter une syntaxe XML pour représenter le schéma lui-même, ce qui permet de réaliser une économie en termes d'outils pour éditer, manipuler et stocker les schémas, puisque ceux-ci deviennent eux-mêmes des documents XML.

Type de données : Les DTDs fournissent seulement un système sommaire de types, basé sur les éléments textuels. Les schémas XSD prolonge ce mécanisme avec une gamme étendue de types, du SQL, du Java tels que *date/heure*, *binaire*, *URI*, etc. De plus, les schémas XSD intègrent la notion d'héritage soit par restriction soit par extension des définitions de types.

Espace de noms : Les espaces de noms permettent à des éléments ayant le même nom d'être employés dans différents contextes. En outre, les types des schémas et leurs

éléments peuvent être inclus (ou importés) dans un schéma XSD séparé en utilisant un même (ou un autre) espace de noms.

Contraintes : Contrairement aux DTDs, les schémas XSD augmentent la richesse des contraintes exprimables notamment des contraintes portant sur les contenus et le type des données : les contraintes de clé et d'unicité, les références (clés étrangères), les contraintes de valeurs (types énumérés), les contraintes de cardinalités et de valeurs par défaut ou optionnelles.

Il semble donc nécessaire d'avoir un modèle unifié des modèles semi-structurés qui essaye de diminuer le fossé sémantique entre eux, mais adaptable aux spécificités de chacun.

3 Le modèle *CRP*

Dans le modèle *CRP*, il existe trois types d'*éléments* de base, à savoir :

- Concept,
- Relation,
- Propriété.

Le modèle *CRP* fait une distinction précise de ces trois éléments. Il spécifie également les contraintes sur les concepts et sur leurs relations telles que le degré et le type des relations, ainsi que les contraintes sur les différentes propriétés. Toutes ces informations nous permettront par la suite d'une représentation formelle de ces éléments.

Définition 1 (*CRP*)

Le modèle *CRP* est représenté par le triplet $\langle C_{CRP}, R_{CRP}, P_{CRP} \rangle$ où :

$\boldsymbol{C_{CRP}}$: est l'ensemble des concepts du schéma modèle,

$\boldsymbol{R_{CRP}}$: est l'ensemble des relations existantes entres les concepts du modèle,

$\boldsymbol{P_{CRP}}$: est l'ensemble des propriétés appartenant aux concepts et aux relations.

EXEMPLE – La figure 2 formalise l'exemple de la figure 1 par notre modèle *CRP*. Dans cette figure, des concepts, des relations et des propriétés apparaissent. L'identifiant de *département* est *nom* (indiquée par un rond noir) et *département* possède aussi une autre propriété *localisation*. Le losange, liant *département* et *cours*, indique qu'il existe une relation *(hiérarchique de type parent/child)* entre ces deux concepts. L'identifiant de *cours* est *code*, il possède aussi une propriété *libelle* qui n'est pas obligatoirement unique. Chaque *étudiant* possède un identifiant *(numéro)*, un *nom* et une *adresse*. La relation entre *cours* et *étudiant* porte le nom *cours_étudiant* notée par abréviation *ce*.

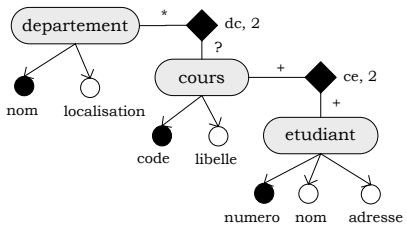


Fig. 2. Représentation d'une structure de données par le modèle *CRP*.

3.1 Concept

Un concept dans le modèle *CRP* est équivalent à la notion d'entité dans le modèle Entité-Relation, de classe dans le modèle objet, et d'élément dans les documents XML. Traditionnellement, un concept est caractérisé par son nom et un ensemble de propriétés que nous définirons plus précisément à la section 3.3. Un concept est lié à un ou à plusieurs concepts via des relations. Nous représentons un concept dans le modèle *CRP* par un rectangle portant son nom.

Définition 2 (Concept)

Un concept $c \in C_{CRP}$ est représenté par le doublet $\langle c_{nom}, c_{contenu} \rangle$

$\boldsymbol{c_{nom}}$: est le nom du concept,

$\boldsymbol{c_{contenu}}$: est le contenu du concept.

Où $P_{CRP}(c)$, $C_{CRP}(c)$ sont respectivement l'ensemble des propriétés et des concepts imbriqués dans le concept c, avec $P_S(c) \cup C_S(c) = c_{contenu}$.

EXEMPLE – Nous considérons l'exemple où chaque *étudiant* possède un *numéro*, un *nom*, une *adresse* et *classe*. Le *numéro* est son identifiant. La figure 3 représente le concept étudiant, avec ses propriétés dans le modèle *CRP*. Donc, on aura $c_{nom} = \langle étudiant \rangle$ et $c_{contenu} = \langle nom, adresse, classe \rangle$.

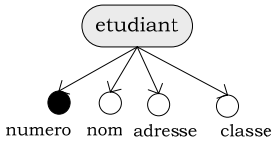


Fig. 3. Représentation du concept *étudiant*.

3.2 Relation

Deux ou plusieurs concepts peuvent être connectés par une relation. Une relation dans *CRP* représente une interconnexion (ou association) entre deux ou plusieurs concepts. Chaque relation est caractérisée par son nom[1]. Elle possède également un degré qui indique le nombre de concepts qu'elle relie. Une relation de degré 2 (e.g. relation binaire) relie deux concepts, l'un des deux est le parent et l'autre est le fils. Les contraintes d'occurrences (ou cardinalités) sont définies dans les relations par la notation issue des modèles objets à savoir *?: zéro-à-un*, ***** *: zéro-à-plusieurs* et + *: un-à-plusieurs*. Une relation est représentée par un losange noir.

Définition 3 (Relation)

Une relation $r \in R_{CRP}$ est représentée par le doublet $\langle r_{nom}, r_{liaison} \rangle$

$\boldsymbol{r_{nom}}$: est le nom de la relation,

$\boldsymbol{r_{liaison}}$: est l'ensemble des concepts reliés via *r*.

EXEMPLE – La figure 4 montre deux relations binaires entre le concept *projet* et le concept *membre*, et entre *membre* et *publication*. La relation entre *projet* et *membre* est notée *« pm, 2, +, + »* signifiant :

pm : son nom *(projet_membre)*,

2 : son degré (entre les deux concepts *projet* et *membre*),

+ : un *projet* concerne au moins un *membre* voire plusieurs,

+ : un *membre* participe à au moins un *projet* voire plusieurs.

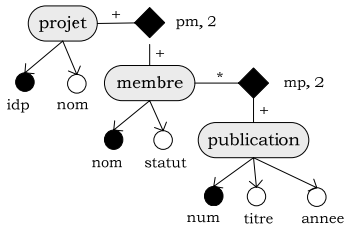


Fig.4. Représentation des relations *implicites* dans le modèle *CRP*.

[1] Parmi les exemples illustrés dans cet article, les noms des relations sont abrégés afin de simplifier la schématisation des figures en concaténant les premières lettres de chaque concept intervenant dans la relation.

Les relations vues précédemment, définissent les relations *implicites* existantes dans les sources de données semi-structurées (e.g. relation d'hiérarchie et d'imbrication entre un *concept-père* et un *concept-fils*). Pour les relations *explicites* (structurelles de type association) entre les concepts (e.g. relation *ID/IDREF*, *key/keyref*) dans les schémas XML. Ces relations sont bidirectionnelles[2], c'est pour cela que nous les représentons par une paire d'arcs (pointillés) reliant les deux concepts en question.

EXEMPLE – Dans la figure 5, les deux concepts *article* et *livre* sont liés au concept *bibliothèque* via deux relations implicites (*ba* et *bl*). Cependant, il existe une relation *explicite* entre ces deux concepts *article* et *livre* spécifiant une association.

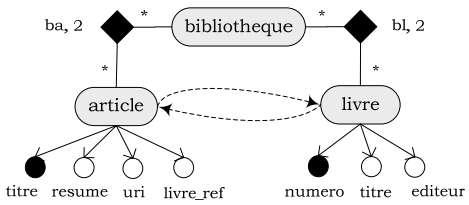


Fig. 5. Représentation des relations *explicites* dans le modèle *CRP*.

3.3 Propriété

Une propriété dans le modèle *CRP* est équivalente à la notion d'attribut dans le modèle relationnel, objet, ou semi-structuré. Selon le schéma d'origine, les propriétés dans *CRP* peuvent apparaître dans les concepts et/ou les relations. Sachant qu'une des différences majeures entre les bases de données relationnelles et les données semi-structurées est la contrainte de l'atomicité des attributs (cf. 1ère forme normale [6]), nous donnons la possibilité de représenter une propriété par :

- clé candidate : attribut à valeur unique pour toutes les instances du concept ;
- clé composée : ensemble d'attributs définissants une valeur unique ;
- clé primaire : parmi les clés candidate ;
- propriété mono-valuée : possédant une seul valeur ;
- propriété multivaluée : possédant un ensemble de valeurs ;
- propriété à valeur required /optional / fixed/ ou default ;
- propriété à structure inconnue : possédant différentes structures dans le schéma.

[2] Afin de garder la compatibilité avec les relations implicites existantes et donc la cohérence dans le modèle *CRP*.

Dans le modèle *CRP*, nous classifions deux classes de propriétés, à savoir :

— *Propriété clé :* qui est représentée à partir d'une clé dans le schéma source ;

— *Propriété simple :* qui n'est pas une propriété clé dans *CRP* et qui représente les cas *mono-valuée, multi-valuées* et *structure inconnu.*

Une propriété est représentée par un cercle étiqueté par son nom. De plus, si cette propriété est une clé (propriété clé), son cercle est noir. Si elle possède une structure irrégulière (multi-structure), le cercle est discontinu. Si la propriété est multi-valuée, nous rajoutons le signe * à l'intérieur du cercle. De même, en fonction des contraintes qu'elle porte, nous rajoutons *R* pour *required* , *O* pour *optional*, *F* pour *fixed*, *D* pour *default*.

Définition 4 (Propriété)

Une propriété $p \in P_{CRP}$ est représentée par le triplet $\langle p_{nom}, p_{catégorie}, p_{domaine} \rangle$

$\boldsymbol{p_{nom}}$: est le nom de la propriété,

$\boldsymbol{p_{catégorie}}$: est le type de la propriété (clé ou simple),

$\boldsymbol{p_{domaine}}$: est le domaine de définition de la propriété.

Où $P_{clé}(c)$ et $P_{simple}(c)$ sont respectivement l'ensemble des propriétés clés et propriétés simples du concept *c*, avec $P_{clé}(c) \cap P_{simple}(c)=\varnothing$ et $P_{clé}(c) \cup P_{simple}(c)= P_{CRP}(c)$.

EXEMPLE – Nous considérons le modèle *CRP* de la figure 6. Le concept *cours* possède les propriétés *code* (propriété clé), *libellé* (propriété simple) et une autre propriété avec une structure irrégulière ou multiple *(info.)*. Le concept *étudiant* possède aussi une propriété clé définit par son *numéro*. Le *nom* de l'*étudiant* qui doit être toujours renseigné ou requis ; en outre, *hobbies* est une propriété à valeurs multiples (un *étudiant* peut avoir plusieurs *hobbies*).

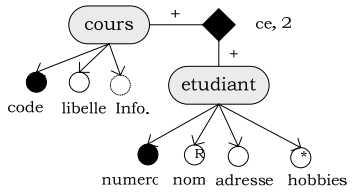


Fig. 6. Représentation des propriétés dans le modèle *CRP*.

Visuellement, on peut remarquer que le modèle *CRP* ressemble au modèle Entité-Relation mais n'est pas identique. Cette représentation à l'avantage de paraître

familière à la communauté des bases de données relationnelles du fait que le modèle ER est un modèle connu ; aussi, notre modèle *CRP* est aisément compréhensible et facilement exploitable. Un autre avantage du modèle *CRP* est qu'il est capable de représenter à la fois les différentes imbrications (hiérarchies) dans les données semi-structurées De plus, il peut modéliser des attributs multi-valués et/ou à structure inconnue qui ne peuvent pas être exprimés dans les autres modèles de données.

4 Mapping entre XSD/*CRP* : Implémentation et validation

Tout au long de ce papier, nous avons présenté le modèle *CRP*. Dans cette section, nous exposons brièvement comment les informations sémantiques dans le modèle *CRP* peuvent être dérivées et utilisées. Premièrement, nous montrons comment les schémas de type XSDs peuvent être représentés dans *CRP*. Dans [18], nous avons défini un ensemble de règles de mapping entre les schémas XML et le modèle *CRP*.

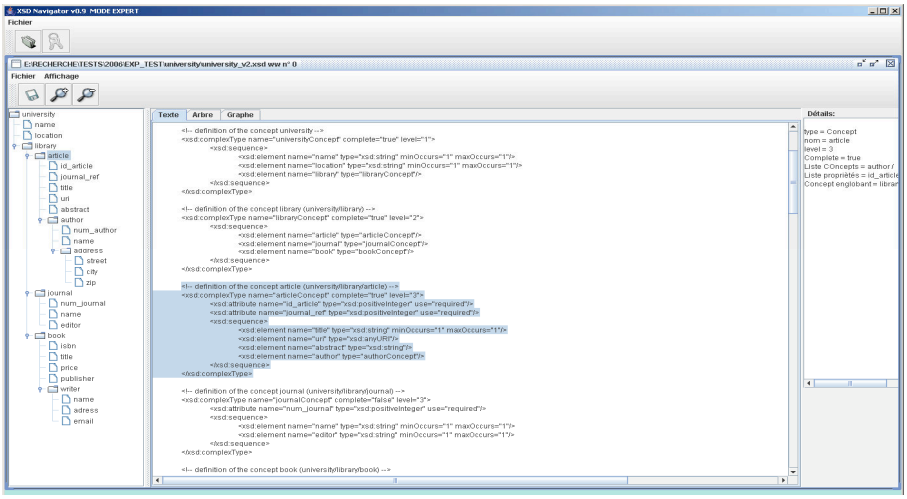
Premièrement l'élément racine *(root element)* du schéma XSD est défini par un concept dans le modèle *CRP*. Ce concept représente l'agrégation ou le *concpet-racine* de tous les autres concepts du modèle. Les éléments XSDs de type *xsd:element* possédant d'autres éléments ou attributs *(xsd:attribute)*, sont représentés par des concepts dans le modèle *CRP*. Pareil pour les éléments dont la structure est complexe *(xsd:complexType)* sont représentés par des concepts. Les éléments de type *xsd:attribute* représentent les propriétés composées. Ainsi que les éléments avec le type *xsd:key* représentent les propriétés clés, tandis que les éléments de type *xsd:keyref* indiquent une référence entre concepts. Aussi les relations sont dérivées à partir des relations hiérarchiques (implicites), et les différentes associations (explicites) existantes. Les cardinalités des éléments et des attributs peuvent être aussi représentées à partir des schémas XML.

Pour des besoins spécifiques, nous représentons le modèle *CRP* avec deux représentations différentes à savoir : représentation arborescente *(TREE)* (voir figure 8(b)) et représentation graphique *(GRAPH)* (voir figure 8(c)). Dans un premier temps nous chargeons les fichiers XSDs à transformer dans notre système[3] (voir figure 8(a)). Pour la génération automatique des modèles *CRP*[4], nous récupérons l'ensemble des informations structurelles et sémantiques propres aux différents éléments du schéma XSD, ainsi que toute son arborescence. L'avantage de ces représentations permet de mieux visualiser, parcourir et comparer des schémas XML hétérogènes, sans recourir à la syntaxe elle-même, qui peut être incompréhensible par certains utilisateurs. Via le modèle *CRP* généré, le concepteur ou l'expert du domaine peut modifier, supprimer, ou mettre à jour directement des informations sur les concepts, les relations et les propriétés du modèle (e.g., modifier le nom d'un concept, les cardinalités ou le

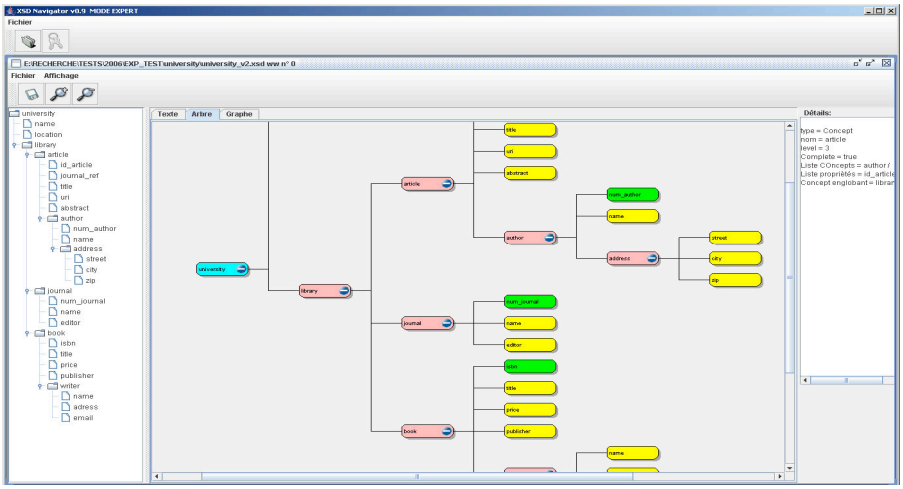
[3] Les schémas XSDs passent préalablement par une phase de *parsing* afin de vérifier leur validité. Pour cela, nous intégrons dans notre système un parseur validateur. Ce dernier se lancera automatiquement dès qu'un schéma XSD est chargé dans le système.

[4] Nous montrons exclusivement ici le mapping XSD/CRP. Notre système a été conçu pour transformer divers schémas semi-structurées tels que, DTD, DSD.

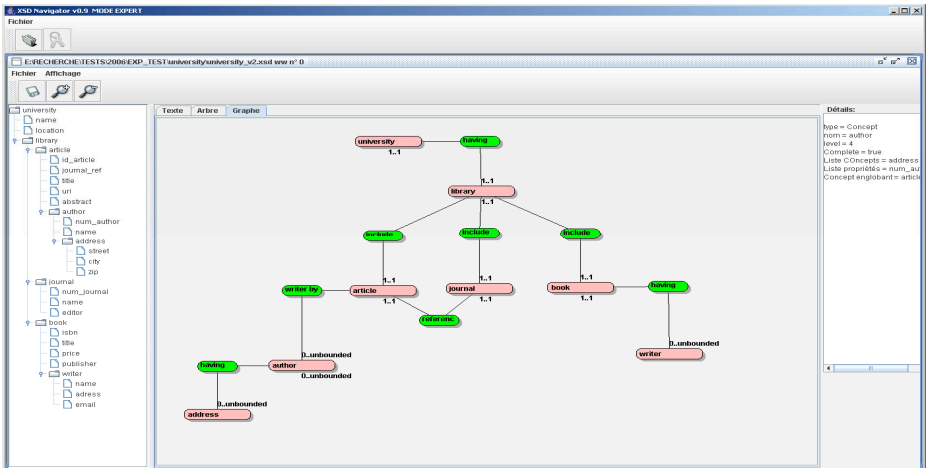
domaine de définition d'une propriété, etc.). Ces modifications consistent à l'enrichissement sémantique [17] semi-automatique des modèles *CRP* dans le but de pallier à la pauvreté sémantique des sources de données semi-structurées.



(a) Aperçu d'un schéma XSD chargé dans notre système *(vue "TEXT")*



(b) Aperçu d'un modèle *CRP (vue "TREE")*



(c) Aperçu d'un modèle *CRP (vue "GRAPH")*

Fig. 8. Génération automatique des modèles *CRP* à partir des schémas XSD.

5 Conclusion

Jusqu'à l'avènement du langage XML, le monde de l'information se divisait en deux parties complémentaires. D'un côté, il y avait le monde des bases de données traditionnelles (hiérarchiques, relationnelles, objets), de l'autre, le monde de l'EDI permettant l'échange des informations et des présentations. La plupart des systèmes d'intégration actuels utilisent XML comme modèle commun. Sa structure irrégulière permet de représenter toutes formes de données, mais présente l'inconvénient de complexifier les algorithmes d'intégration des données de différentes sources, notamment l'intégration des schémas.

Le travail présenté dans cet article s'insère dans le cadre des travaux liés à l'intégration de données et plus particulièrement à l'intégration de schémas. Nous avons défini un modèle *CRP* par l'utilisation d'un ensemble minimal de type d'éléments (concept, relation, propriété), par l'emploi d'une notation claire et facilement compréhensible, et par sa capacité à représenter à la fois la structure et la sémantique des schémas hétérogènes. L'un des objectifs du modèle *CRP* est de diminuer l'ambiguïté sémantique entre les différents types d'éléments.

Le modèle *CRP* peut être utilisé dans plusieurs voies de recherches telles que, le stockage des structures issues des BDs XML natives, la conception de vues XML, l'évaluation des requêtes, etc. Dans le cadre de nos recherches, nous exploitons le modèle *CRP* pour la définition des schémas XML riche en sémantique, dans le cadre de l'intégration de schémas sources hétérogènes. Dans un tel contexte, partager et réutiliser des ressources existantes s'avère d'un intérêt incontestable, à la fois en termes d'interopérabilité entre les applications et entre les différents utilisateurs.

Nous envisageons dans le futur, améliorer ce modèle en tenant compte d'autres contraintes telles que les dépendances fonctionnelles, les formes normales, l'héritage, etc. afin d'enrichir sémantiquement le modèle *CRP*. Ce qui permettra (i) d'étendre ce formalisme conceptuel sur d'autres modèles tels que le relationnel ou même l'objet, et (ii) de mieux représenter les schémas sources afin de produire des *matching/mapping* corrects entre les ces derniers.

Références

1. Abiteboul, S., Buneman, P., Suciu, D.: Data on the Web: From Relations to Semistructured Data and XML, Morgan Kaufmann, (1999)
2. Bergamaschi, S., Castano, S., Vinci, M.: Semantic integration of semistructured and structured data sources, SIGMOD Record, (1999) 54-59
3. Bray, T., Paoli, J., Sperberg-McQueen, C.M., Maler, E.: Extensible markup language (xml) 1.0, second edition. http://www.w3.org/TR/2000/REC-xml-20001006, (2000)
4. Buneman, P., Susan, B., Davidson, D., Fernandez, M.F., Suciu, D.: Adding structure to unstructured data, In Proceedings of the 6th International Database Theory Conference Theory (ICDT'97), (1997) 336-350
5. Cluet, S., Delobel, C., Simeon, J., Smaga, K.: Your mediators need data conversion!, In Proceedings ACM SIGMOD International Conference on Management of Data, ACM Press, (1999) 177-188
6. Codd, E.F.: Relational Completeness of Databases Sublanguages, Prentice Hall, Englewood, Cliffs, NJ, (1972)
7. Fernandez, M., Florescu, D., kang, J., Levy, A., Suciu, D.: Catching the boat with Strudel: experiences with a web-site management system. In SIGMOD, (1998) 414-425
8. Garcia-Molina, H., Papakonstantinou, Y., Quass, D., Rajamaran A., Sagiv Y., Ullman, J., Vassalos, V., Windom, J.: The TSIMMIS Approach to mediation: Data Models and Languages. Journal of Intelligent Information Systems, (1997)
9. Goldman, R., Widom, J.: Enabling query formulation and optimization in semistructured databases, In Proceedings of 23rd International Conference on Very Large Data Bases (VLDB'97), Morgan Kaufmann, (1997) 436-445
10. Lotus Development Corporation. Lotus development corporation. http://www.lotus.com/
11. Luddsher, B., Himmervder, R., Lausen, G., May, W, Schlepphorst, C.: Managing semistructured data with FLORID: a deductive object-oriented perspective. Information Systems, (1998) 589-613
12. McHugh, J., Abiteboul, S., Goldman, R., Quass, D., Widom, J.: Lore: A database management system for semistructured data. SIGMOD Record, (1997) 54-66
13. Shanmugasundarm, J., Tufte, K., Zhang, C., He, G., DeWitt, J., Naughton J.F.: Relational databases for quering XML documents: Limitations and opportunities. In Proceedings VLDB'99, Morgan Kaufmann, (1999) 79-90
14. Tamino – An Internet Database System. http://www.tamino.com/
15. W3C-XML, Extensible Markup Language (XML) 1.0, W3C Recommendation, http://www.w3.org/TR/REC-XML, (1998)
16. W3C-XSD, XML Schema Part 0: Primer, W3C Recommendation, 2001, http://www.w3.org/TR/XmlSchema-0/, (2001)
17. Zerdazi, A., Lamolle, M.: Hyperschema XML: Un modèle d'intégration par enrichissement sémantique de schémas XML, MajecSTIC'05, (2005) 143-150
18. Zerdazi, A., Lamolle, M.: Modélisation des schémas XML par adjonction de métaconnaissances sémantiques, ASTI'05, (2005) 29-32

Une approche basée sur les services web utilisant les spécifications ebXML pour e-commerce B2B

A.Raouf Kouhoul[1], Fethi Rabhi[2] and Ryme Chelouah[3]

[1]Université A.Mira de Bejaia, Département d'Informatique, Bejaia 06000, Algérie
Kouhoul_abderaouf@yahoo.fr
[2]University of New South Wales, School of information Systems and Management,Sydney NSW 2052, Australia f.rabhi@unsw.edu.au

[3]Université A.Mira de Bejaia, Département d'Informatique, Bejaia 06000, Algérie
ryme_chelouah@yahoo.fr

Abstract. The introduction of the meta-language XML (eXtensible Mark-up Language) has a great impact on the domain of interchanges B2B. Several standards based on XML have been proposed such as ebXML, RosettaNet, Biztalk, eCo...

In this article, we try to give the different steps to achieve a project of interchange of business documents B2B based on the ebXML standard. First, we start by giving a survey about EDI technologies, their advantages, limits and the steps of an EDI project. Then, we introduce shortly the ebXML standards; their goals, architecture. So, similarly to the EDI, we have proposed a merely methodology to set up an ebXML project basing on previous experiences on EDI project. For each step, we have précised the EDI elements which may be re-used, new features,
and issues.

Keywords: e-commerce, B2B, XML, ebXML…

1 INTRODUCTION

XML, le langage fondateur de plusieurs technologies, est le résultat de la coopération d'un grand nombre d'entreprises et de chercheurs partenaires de W3C-World Wide Web Consortium. XML pallie aux limites de HTML en permettant de séparer le contenu d'un document de sa structure et de sa présentation [3]. Les documents XML sont auto descriptifs: ils permettent d'ajouter la description des données sous forme de méta données. C'est un standard reconnu et exploité par plusieurs partenaires commerciaux. Cependant XML et ses outils ne sont pas suffisants pour permettre l'échange de documents. En effet l'atteinte d'une interopérabilité entre entreprises et d'une sémantique requiert une spécification universelle. C'est dans ce cadre que l'ebXML a été proposé [2].

2 Commerce Electronique

L'émergence de l'Internet a une portée véritablement révolutionnaire. L'Internet ne nécessite que peu d'investissements spécifiques, est d'un coût d'utilisation modeste et est accessible au grand public. Sa relative indépendance par rapport à la nature des infrastructures, aux frontières et aux opérateurs lui a permis un développement extrêmement rapide. Cette croissance a donné naissance à plusieurs nouveaux domaines d'application de nouvelles technologies d'information et de communication (TIC) comme e-commerce, e-learning,

e-gouvernment...

De nos jours, l'Internet offre des nouvelles technologies qui ont les potentialités de révolutionner le monde des affaires. En effet, grâce à son coût réduit et sa rapidité de diffusion, l'Internet permet à des partenaires commerciaux d'échanger des informations d'affaire ; il s'agit du commerce électronique. Le commerce électronique peut se définir comme la rencontre, sur un réseau international de télécommunication, d'une offre et d'une acceptation qui constitue un contrat de vente, de prestation de service, ou pour toute opération commerciale entre deux partenaires (vendeur, acheteur, fournisseur, prestataire de services [8].

Au cours de son développement, le commerce électronique s'est manifesté dans quatre grands domaines [8]:

- **Le B to B-Business to Business** : Concerne les transactions électroniques entre deux ou plusieurs entreprises. C'est le domaine par excellence du commerce électronique; 98% des échanges dématérialisés se font entre entreprises.
- **Le B to C-Business to Consumer** : Transaction entre une entreprise et une personne privée. C'est de ce type de commerce dont on parle dans la presse. Mais en réalité le taux d'utilisation reste très faible même si les potentiels de croissance sont très importants. Les freins sont encore nombreux en particulier vis-à-vis des moyens de paiement à distance.
- **Le B to A-Business to Administration**: Transaction électronique entre une entreprise et une administration, par exemple pour la transmission d'une déclaration des impôts.
- **Le A to C-Administration to Consumer**: Transaction électronique entre administration et une personne physique.

3 EDI

L'EDI, Electronic Data Interchange, est un outil au service de l'échange électronique consistant à transporter automatiquement de l'application informatique d'une entreprise à l'application informatique d'une autre entreprise, par des moyens de télécommunication, des données structurées selon des messages types convenus à l'avance [8]. De nos jours, L'EDI est la solution la plus utilisée pour les échanges de type B2B dans plusieurs secteurs industriels tels que l'industrie automobile en Europe, le transport en Amérique et les semi-conducteurs en Asie.

3.1 Avantages et limites de l'EDI

L'EDI n'est pas à la portée des PME/PMI à cause du coût de sa mise en oeuvre qui nécessite l'usage des réseaux RVA assurant les fonctions d'enveloppe, le routage, la sécurité. En plus, il nécessite des accords d'interchange qui définissent l'aspect juridique et les aspects techniques. Pour toutes ces raisons, EDI est préférable pour les échanges réguliers et nombreux entre partenaires stables [4].

3.2 Mise en oeuvre d'un projet EDI

La mise en oeuvre d'un projet d'échange électronique EDI nécessite une restructuration des ressources techniques et humaines de l'entreprise. Il nécessite un effort plus

qu'organisationnel et économique que technique. En effet, il est indispensable d'opter pour une approche méthodologique en spécifiant pour chaque étape les objectifs, les outils. Dans ce qui suit, on présente la méthodologie proposée par les organisations EDIFRANCE et GENCOD EAN France [5]. La figure 1 illustre l'enchaînement des différentes étapes de mise en oeuvre d'un projet EDI qui est un processus séquentiel complexe, organisationnel, technique, commerciale et législatif. Le modèle illustré par la figure 1 ressemble dans sa totalité au processus de génie logiciel en cascade. La description de ces étapes et les outils qui y sont utilisés sont donnés dans le Tableau1. Pour mener à bien un projet EDI il faut concilier des compétences informatiques, économique, organisationnelles et juridiques.

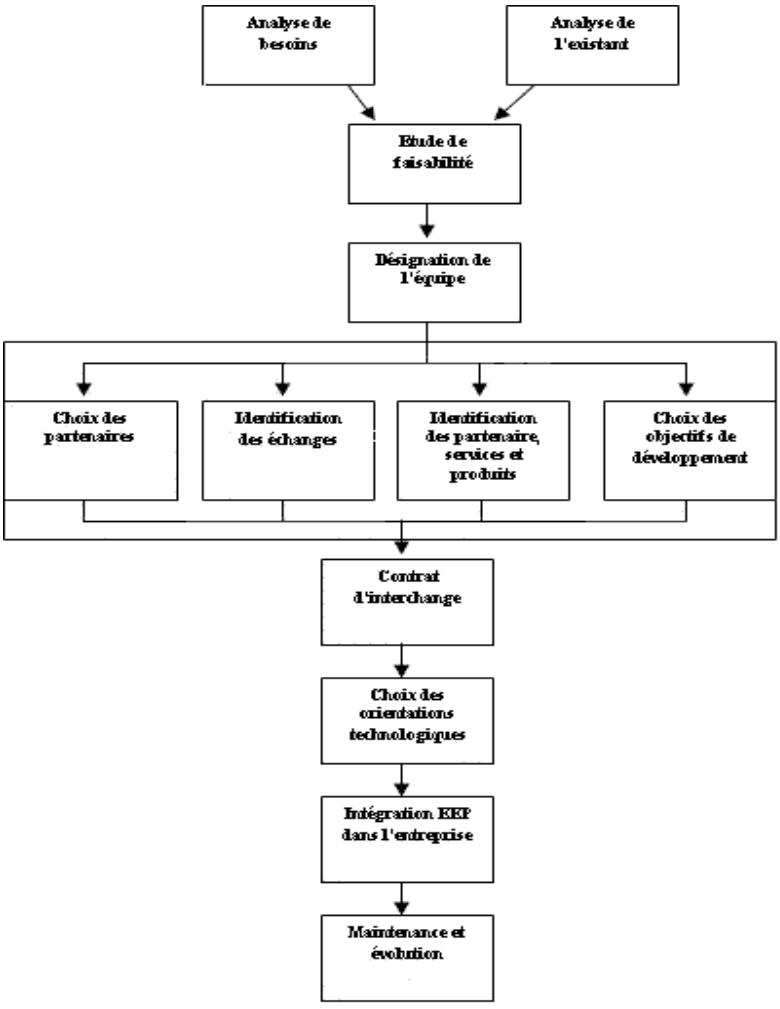


Fig. 1. Etapes de mise en oeuvre d'un projet EDI.

1. **Analyse des besoins:** Effectuer une analyse macro-économique et analyser les stratégies EEP des principaux partenaires de l'entreprise pour identifier avec précisions les besoins en matières d'échanges EEP et de spécifier les flux dématérialisable.

2. **Analyse de l'existant:** La deuxième étape consiste à observer la structure interne de l'entreprise pour synthétiser les processus organisationnels et administratives ainsi que les données, les interfaces externes/internes de systèmes d'informations applicatifs et les ressources humaines/matérielles dont dispose l'entreprise. Pour mener à bien cette étape, on fait recours aux différentes techniques de recueil de données, d'enquêtes et d'observation. Les diagrammes UML, diagramme de cas d'utilisation, diagramme de séquence, diagramme de classes, peuvent être utilisés pour modéliser les flux de données.

3. **Etude de faisabilité:** En fonction des besoins identifiés, l'état courant de l'entreprise, la volonté et la capacité d'investissement déterminer les objectifs et préciser les contraintes techniques et fonctionnels du projet, temps réalisation, coût, etc.

4. **Désignation de l'équipe EDI:** L'équipe EDI doit comporter des décideurs, des informaticiens, des juristes, des économistes. Cette équipe à pour mission l'établissement d'un plan d'action.

5. **Analyse des flux d'échange:**Cette étape résulte en:

- Choix des partenaires qui peut se faire selon plusieurs critères, par exemple en se basant sur les relations déjà existantes, le volume de lignes de commandes, etc.
- Identification des scénarios d'interchange qui consiste en identification des partenaires, des services et des
  produits et qui pourra ensuite raffiner la description des objectifs.

6. **Etablissement des contrats d'interchange :** Cette étape consiste en établissent d'une convention d'interchange, définition de la couverture juridique du contrat, redéfinition des relations commerciales et réglementation législative qui couvrent l'activité concernée.
7. **Choix des orientations technologiques:** Les partenaires devront choisir le standard, format de leurs messages, la version du langage, la fréquence d'échange, le moyen de communication et le paramétrage de la station EDI. Les partenaires utilisent les guides d'utilisation des messages et les guides utilisateur de la station EDI.

8. **Intégration des EEP dans l'entreprise:** L'intégration des EEP implique des tests d'intégration- jeu de tests, formation des utilisateurs et des évaluations, support de formation, guides d'utilisation des messages et guides utilisateur de la station EDI.

9. **Maintenance et Evolution:** Consiste en suivi et contrôle des EEP et la réingénierie des scénarios d'interchange.

4 EBXML

L'ebXML est un standard promu par l'UN/CEFACT, United Nations Center For Trade Facilitation And Electonic Business, et OASIS- Organization for the Advancement of Structural Information Standard. Il met en place une plate-forme destinée au commerce électronique, permettant à des entreprises de se rencontrer et de travailler ensemble en utilisant des messages XML, dans le cadre de processus métiers standardisés ou mutuellement négociés par les partenaires [6]. Sa fonction première est de spécifier des règles de conception pour décrire des processus B2B. A savoir, la description des processus d'échange entre partenaires, gestion des propositions commerciales, appels d'offres, etc. L'ebXML inclut également ses propres spécifications d'inscription et d'annuaire de services Web [10]. Les composants de l'ebXML sont illustrés dans la figure 2.

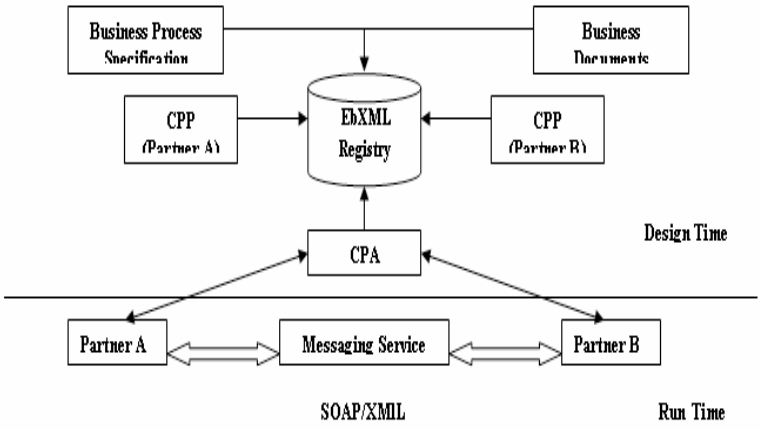


Fig. 2. Composants de l'architecture ebXML.

Le standard ebXML propose des solutions techniques à la mise en oeuvre de relations commerciales pour les grandes entreprises ainsi que pour les entreprises de financement faible. De point de vue méthodologique, mettre en oeuvre un projet ebXML ressemble dans sa totalité aux projets EDI ; les mêmes parties des entreprises sont impliquées, la structure organisationnelle et les relations interentreprises demeurent stables. Cependant, l'entreprise a plus de chances de découvrir de nouveaux partenaires. La différence se trouve dans l'utilisation des CPP et des CPA pour faciliter la découverte de nouveaux partenaires ainsi que l'élaboration des contrats d'interchange.

5 Problématique de la migration EDI vers ebXML

L'ebXML est envisagé comme une solution adéquate qui permettra à n'importe quel partenaire, que ce soit une grande entreprise ou PME/PMI, d'établir des relations d'affaires et d'intégrer les transactions B2B. Cependant, le passage de l'EDI à l'ebXML ne se réduit pas à un simple remplacement d'environnements logiciels. Pour les entreprise n'utilisaient

pas l'EDI, il leur faudrait de redéployer leurs ressources et de s'investir dans un projet de mise en œuvre d'une plate forme ebXML en suivant une démarche similaire à celle de l'ebXML. De l'autre coté, pour les entreprise ayant de l'expérience dans les technologies EDI, cette transition nécessite la détermination préalable d'une démarche de migration pour mener à bien le progrès vers la reconfiguration des réseaux de partenaires autour des relations d'affaires dynamiques, configurables et personnalisables. Il va falloir définir comment exploiter le savoir-faire et la performance acquis dans la pratique de l'EDI. Cette démarche doit couvrir les aspects technologiques et les aspects organisationnels.

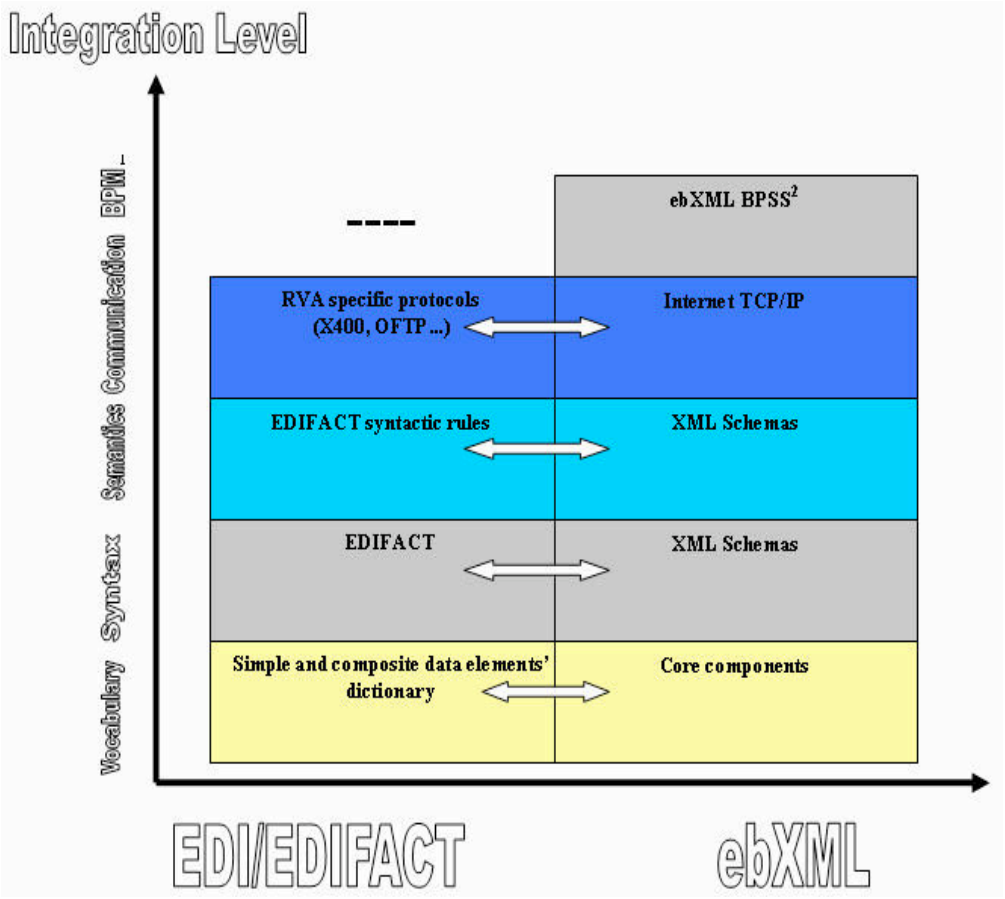


Fig. 3. Niveaux d'intégration des transactions B2B

A chaque niveau d'intégration des transactions B2B (Fig.3), il est nécessaire de définir comment passer de l'EDI à l'ebXML et en plus comment assurer la co-existence des deux dans la même communauté d'interchange. Absolument, des solutions et des réponses devront être apportées à la problématique de migration, ce qui est l'objet de plusieurs travaux de recherche .

[1] BPM : Business Process Management
[2] BPSS : ebXML Business Process Specification Schema

6 Implémentation de l'architecture ebXML basée sur les services Web

Une application conforme aux spécifications ebXML peut être implémentée sous forme d'un web service. En effet, les Web Services sont devenus le nouveau point de convergence technologique de l'ensemble des acteurs du marché de l'informatique. L'objectif des Web Services est de faciliter l'accès aux applications entre entreprises et ainsi de simplifier les échanges de données. Les principales caractéristiques des web services sont [1]:

- Fournir une interface à travers laquelle ils pourront être invoqués ; accessibles par le web.
- Pouvoir être publié et localiser à travers un enregistrement. Pouvoir interagir avec d'autres web services.
- Etre modulaires et réutilisables.

L'utilisation des web services comme un modèle architecturale pour l'implémentation de l'architecture proposée offre des nouvelles perspectives, aux acteurs du domaine de e-commerce, allant de la réutilisation des applications internes grâce au couplage faible et à la séparation fonctionnelle entre l'interface d'utilisation de chaque web service enveloppant un module de l'architecture et de son implémentation. En plus, l'utilisation du protocole SOAP assure l'interopérabilité entre les différents modules.

La réutilisation et l'interopérabilité sont deux exigences des architectures évolutives. Cependant, de point de vue fonctionnel et organisationnel, les entreprise (grandes et PME/PMI) cherchent des solutions logicielles pour augmenter leur productivité, efficacité, efficience, agilité et réactivité. Cette architecture basée web services aura un impact sur chaque aspect ; une étude empirique est nécessaire pour quantifier et qualifier cet impact. Néanmoins, à partir des limites des applications actuels et des attentes des entreprises, on peut résumer les apports de l'architecture par rapport aux solutions EDI en :

- Utilisation des technologies Internet accessible à un coût inférieur au coût des solutions EDI.
- Moins d'investissement technologique, plus de productivité.
- Facilité de personnalisation et de configuration des relations d'affaires « dynamic businesss networks »
- Etc.

Dans cet article, nous proposons l'architecture schématisée par la figure 4 :

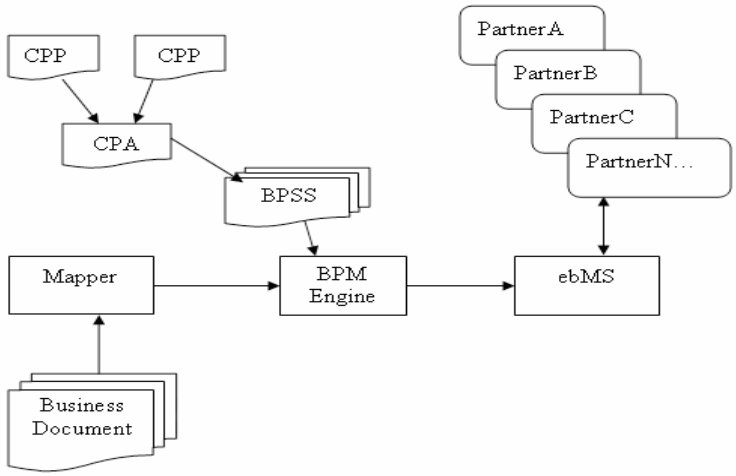


Fig. 4. Architecture ebXML.

L'architecture proposée se compose de trois modules et utilise quatre types de fichiers de paramétrage.

1. **Mapper :** responsable de la conversion des documents d'affaire parvenant des systèmes applicatifs internes formatés suivant un format spécifique au partenaire d'affaire vers un autre format standard sectoriel ou international. Généralement, c'est le module le plus difficile à développer est cela est du à la complexité des formats de documents d'affaires.

2. **BPM Engine :** Les fonctions principales de ce module sont :
- Assurer la coordination entre les modules de l'architecture.
- Contrôler et suivi de processus d'affaire décrit par le document BPSS.

3. **ebMS :** L'interface de communication de l'architecture avec les partenaires d'affaire. Ce module s'occupe :
- Etablissement des connexions distantes .
- Envoi et réception des documents d'affaires.

Les trois modules décrits précédemment utilisent les informations contenues dans les fichiers de paramétrage suivants :

- **CPP-Collaboration Protocol Partner [9] :** Ce document spécifie les capacités techniques d'un partenaire lui permettant de participer dans un processus d'affaire.

- **CPA-Collaboration Protocol Agreement [9] :** Ce document résulte de la fusion de deux CPA ; il spécifie l'aspect technique du contrat d'interchange.

- **BPSS-Business Process Spacification Schema [9] :** Ce document décrit le processus d'affaire. ebXML propose un modèle objet pour les spécifiés.

- **Business Documents :** Ce sont les documents d'affaires échangés entre les partenaires d'affaires, commande, facture avis d'expédition,....

Les différents modules interagissent pour intégrer les transactions B2B. Le module ebMS se charge des communications nécessaires aux processus métiers et en plus il assure la fonction d'ajout et de suppression des entêtes (Packing and Unpaking) aux documents d'affaires pour obtenir des structures adéquates à la transmission sur Internet en tenant compte des problèmes de sécurité, confidentialité, intégrité … Les documents d'affaires subissent des transformations et des conversion de formats d'un format interne spécifique au partenaire en un format standard international ou sectoriel, cette fonctionnalité est assurée par le Mapper. Le moteur des processus métier joue le rôle de coordinateur des modules et contrôle l'avancement du processus en cours d'exécution.

CONCLUSION

Dans cet article, nous avons propose une architecture orientée service web pour implémenter les spécifications ebXML; une architecture modulaire, extensible et flexible. Le standard ebXML a été proposé pour assurer l'échange de documents d'affaires ainsi l'interopérabilité entre leurs systèmes applicatifs. Une architecture basée sur la technologie multi-agent a été proposée [02]. L'interopérabilte est un facteur qui renforce le recours aux services web.

Comme extension dans les futurs travaux de recherche, la définition d'une démarche de migration EDI vers ebXML en tenant compte des différents niveaux d'intégration des transactions B2B apparaît nécessaire. Une étude empirique sur un échantillon d'entreprises est nécessaire pour soulever tous les problèmes et les contraintes éventuelles liés à la mise en œuvre d'une plate forme d'intégration des transactions B2B conforme aux spécifications ebXML.

References

[1] B.Benatallah,F.Rabhi and N.Mihandjiev "E-Commerce Enabling Technologies-3'' rd Edition, Australie, January 14 th , 2003.

[2] D.Boughaci and H.Drias ''An agent-based approach using the ebXML Specification for E-business'', In proceeding of the International Conference on Information Technology (ITCC'05), 2005.

[3] H.Kadima and V.MonfortLes services web '' Les services web'', Dunod, 2003.

[4] EDIFrance and GENCOD EAN France'' Comprendre XML pour les échanges électroniques professionnels'', TechRep, Mai 2002.

[5] EDIFrance and GENCOD EAN France'' Le commerce électronique pour l'entreprise'', TechRep, Janvier 2000.

[6] J.Chauvet , "Service Web avec SOAP, WSDL, UDDI, ebXML", Eyrolles, 2002.

[7] Marc Langlois and Stéphane Gasch, "Le commerce électronique B2B De l'EDI à l'Internet", Dunod, 2001.

[8] R.MARCHAND, H.AGNOUX and C.CHIARAMONTI'' Applications EDI sur l'Internet Commerce électronique B2B'', Eyrolles, 1999.

[9] OASIS EBXML Specification.
Available : http://www.ebxml.org

[10] GS1 Specifications.
Available: http://www.gs1.org.

On b-colorings in regular graphs*

Mostafa Blidia† **Frédéric Maffray**‡ **Zoham Zemir**†

February 6, 2007

Abstract

A b-coloring is a coloring of the vertices of a graph such that each color class contains a vertex that has a neighbor in all other color classes. El-Sahili and Kouider have conjectured that every d-regular graph with girth at least 5 has a b-coloring with $d+1$ colors. We show that the Petersen graph infirms this conjecture, and we propose a new formulation of this question and give a positive answers for small degree.
Keywords: Coloration, b-coloring, a-chromatic number, b-chromatic number.

1 Introduction

A proper coloring of a graph $G = (V, E)$ is a mapping c from V to the set of positive integers (colors) such that any two adjacent vertices are mapped to different colors. Each set of vertices colored with one color is a stable set of vertices of G, so a coloring is a partition of V into stable sets. The smallest number k for which G admits a coloring with k colors is the chromatic number $\chi(G)$ of G.

Many graph invariants related to colorings have been defined. Most of them try to minimize the number of colors used to color the vertices under some constraints. For some other invariants, it is meaningful to try to maximize this number. The b-chromatic number is such an example. When we try to color the vertices of a graph, a simple trick consists in starting form a coloring and trying to decrease the number of colors by merging color classes. This motivated the introduction of *a-colorings* and the *a-chromatic number* by Harary, Hedetniemi and Prins [5]. An a-coloring is a proper coloring such that the subgraph induced by any two different color classes contains an edge. Clearly, the process merging suggested above is impossible when we start from an a-coloring. So the a-coloring number is a measure of how hard it is to obtain a coloring with few colors. This inspired Irving and Manlove [7, 8] to introduce another procedure, which consists, given a coloring, in trying to reduce the number of colors by transfering every vertex from a fixed color class to another

*This research was supported by the Algerian/French program CMEP/Tassili 05 MDU 639.

†Département de Mathématiques, Université de Blida, B.P. 270, Blida, Algeria. E-mail: m_blidia@yahoo.fr, zohaze@yahoo.fr

‡C.N.R.S, Laboratoire Leibniz-IMAG, 46 Avenue Félix Viallet, 38031 Grenoble Cedex, France. E-mail: frederic.maffray@imag.fr

color class. A *b-coloring* is a proper coloring such that every color class i contains at least one vertex that has a neighbor in all the other classes. Any such vertex will be called a *b-dominating vertex* of color i. The *b-chromatic number* $b(G)$ is the largest integer k such that G admits a b-coloring with k colors.

Clearly every $\chi(G)$-coloring of a graph G is b-coloring, and so $\chi(G) \leq b(G)$. The behavior of the b-chromatic number can be surprising. For example, the values of k such that a graph admits a b-coloring with k colors do not necessarily form an interval of the set of integers; in fact any finite subset of $\{2, \ldots\}$ can be the set of these values for some graph [3].

Irving and Manlove [7, 8] proved that deciding whether a graph G admits a b-coloring with a given number of colors is an NP-complete problem, even when it is restricted to the class of bipartite graphs. On the other hand, they proved that the problem is polynomially solvable for trees. This NP-completeness results has incited researchers to establish bounds on the b-chromatic number in general or to find its exact values for subclasses of graphs.

For any vertex v of a graph G, the *neighborhood* of v is the set $N(v) = \{u \in V(G) \mid uv \in E\}$ and the *degree* of v is is $\deg(v) = |N(v)|$. A graph G is d-regular if every vertex of G has degree equal to d. For integer $k \geq 3$, we let C_k denote the cycle with k vertices. The *girth* of G is the length of a shortest cycle in G. In [2], El-Sahili and Kouider pose the following question: *Is it true that every d-regular graph G with girth $g(G) \geq 5$ satisfies $b(G) = d + 1$?* We discovered that the Petersen graph offers a negative answer to this question. However, we also propose some positive results, in the case where $d = 3, 4, 5$, which suggest that the Petersen graph might be the only counterexample to the question.

Theorem 1 *The Petersen graph has b-chromatic number* 3.

Proof. Let G be the Petersen graph, with vertices $v_1, \ldots, v_5$, $w_1, \ldots, w_5$ such that v_1, v_2, v_3, v_4, v_5 induce a 5-cycle in this order, w_1, w_3, w_5, w_2, w_4 induce a 5-cycle in this order, and $v_i w_i$ is an edge for each $i = 1, \ldots, 5$. First we show that G has a b-coloring with 3 colors. By assigning color 1 to v_1, v_4, w_2, w_3, color 2 to v_2, w_4, w_5, and color 3 to v_3, v_5, w_1, we obtain a coloring in which v_1, v_2, v_3 are b-dominating.

Now suppose that G admits a b-coloring with 4 colors. For $j = 1, \ldots, 4$, let d_j be a b-dominating vertex of color j, and let $D = \{d_1, \ldots, d_4\}$. Note that each vertex d_j must have exactly one neighbour of each of the three colors different from j.

Suppose that D induces a stable set. So, up to symmetry, we can assume that $D = \{v_1, v_3, w_4, w_5\}$, with $d_1 = v_1$, $d_3 = v_3$, $d_2 = w_4$, $d_4 = w_5$. Without loss of generality, v_2 has color 2, and so w_2 does not have color 2. Since v_1 is b-dominating, it must have a neighbor of color 4, which can only be w_1, and a neighbor of color 3, which can only be v_5. Since v_3 is b-dominating, it must have a neighbor of color 4, which can only be v_4, and a neighbor of color 1, which can only be w_3. But now w_5 cannot have a neighbor of color 2, a contradiction. So D does not induce a stable set.

We may assume that d_1, d_2 are adjacent, and, up to symmetry, that $d_1 = v_1$ and $d_2 = v_2$ (since all edges of the Petersen graph play the same role). Now, it is easy to see that, wherever d_3 may be, there is a C_5 of G that contains d_1, d_2, d_3. So (since all C_5's of the

Petersen graph play the same role), we can assume that d_3 is one of v_3, v_4, v_5. Up to symmetry, this leads to two cases.

Case 1: $d_3 = v_3$. Since v_2 is dominating, it has a neighbour of color 4, which can only be w_2. One of v_4, v_5, say v_5, does not have color 4. Since v_1 is b-dominating, it has a neighbour of color 4, which can only be w_1, and a neighbour of color 3, which can only be v_5. Since v_3 is dominating, it has a neighbour of color 4, which can only be v_4, and a neighbour of color 1, which can only be w_3. Note that w_5 can only have color 2. But now, the vertices of color 4 are w_1, w_2, v_4, and each of them has two neighbours of the same color, so none of them can be b-dominating, a contradiction.

Case 2: $d_3 = v_4$. Since v_1 is b-dominating, it has a neighbour of color 3, which can only be w_1, and a neighbour of color 4, which can only be v_5. Likewise v_2 has a neighbour of color 3, which can only be w_2, and a neighbour of color 4, which can only be v_3. But then v_4 has two neighbours of color 4, so it cannot be b-dominating, a contradiction. □

Theorem 2 ([6]) *Every d-regular graph G with girth $g(G) \geq 6$ has a b-coloring with $d+1$ colors.*

Proof. Let x be a vertex of G, and let $x_1, \ldots, x_d$ be its neighbors. For each $i = 1, \ldots, d$, let $N_i = N(x_i) \setminus \{x\}$. Then each N_i is a stable set, for otherwise G would contain a cycle of length three. Then any two N_i's are disjoint, for otherwise G would contain a cycle of length four, and there is no edge between them, for otherwise G would contain a cycle of length five. Thus we can obtain a coloring with $d+1$ colors $0, 1, \ldots, d$ as follows. Assign color 0 to x, and for $i = 1, \ldots, d$, assign color i to x_i, and assign to the vertices of N_i the colors from $\{1, \ldots, d\} \setminus \{i\}$, in a one-to-one fashion. Finally, color the remaining vertices in arbitrary order, assigning to each v a color from $\{0, 1, \ldots, d\}$ different from the colors already assigned to its neighbors. Clearly, we obtain a b-coloring that uses $d+1$ colors in which the vertices $x, x_1, \ldots, x_d$ are b-dominating. □

2 A new proof of El-Sahili and Kouider's theorem

El-Sahili and Kouider [2] proved the following theorem.

Theorem 3 ([2]) *If G is a d-regular graph with girth $g(G) \geq 5$ and G contains no C_6, then $b(G) = d+1$.*

We propose here a new proof, using a theorem of Gravier [4] on list coloring. Let L be a mapping that assigns to each vertex v of a graph G a set $L(v)$ of admissible colors. An L-coloring is a coloring c of the vertices of G such that $c(v) \in L(v)$ for every vertex v of G. If G admits an L-coloring it is said to be L-colorable. Let l be a mapping that assigns a positive integer $l(v)$ to each vertex v of G. Then G is said to be *l-list-colorable* if it admits an L-coloring for every L such that $|L(v)| \geq l(v)$ for every $v \in V$. Given an integer k, G is

k-list-colorable if it is l-list-colorable for the mapping $l(v) = k$, $\forall v \in V$. The *list-chromatic number* $\chi_L(G)$ is the smallest integer k such that G is k-list-colorable.

Theorem 4 ([4]) *Let G be a connected graph different from a complete graph and from an odd cycle. Then G is Δ-list-colorable, where Δ is the maximum degree in G.*

Proof of Theorem 3. If $d = 2$, then G is a disjoint union of cycles, all of length 5 or more. Then $b(G) = 3$. Indeed, in order to obtain a b-coloring with 3 colors, it suffices to give colors $1, 2, 3, 1, 2$ to five consecutive vertices in one cycle of G, and to color the rest of G with colors $1, 2, 3$. Now we assume that $d \geq 3$.

Let v be a vertex of G, with neighborhood set $N(v) = \{v_1, v_2, \ldots, v_d\}$, and let G_v be the subgraph of G induced by $(N(v_1) \cup \cdots \cup N(v_d)) \setminus \{v\}$. For $i = 1, \ldots, d$, set $E_i = \{xy \mid x, y \in N(v_i) \setminus \{v\}\}$. Then let H_v be the graph obtained from G_v by adding into its edge-set all the elements of $E_1 \cup \cdots \cup E_d$. We claim that:

$$\text{If } \chi_l(H_v) = d - 1, \text{ then } b(G) = d + 1. \tag{1}$$

To prove (1), we define a list assignment L on H_v as follows. If x is any vertex of H_v, we have $x \in N(v_i)$ for some neighbor v_i of v; then assign the list $L(x) = \{1, \ldots, d\} \setminus \{i\}$ to x. Since $\chi_l(H_v) = d - 1$, there is an L-coloring c of H_v. Extend c to a coloring of G as follows. Give color 0 to v; for $i = 1, \ldots, d$, give color i to the i-th neighbor v_i of v; then color the remaining vertices in arbitrary order, giving to each vertex z a color from $\{0, 1, \ldots, d\}$ different from the colors already assigned to its neighbors. Then we obtain a coloring of G with $d + 1$ colors, and clearly the vertices $v, v_1, \ldots, v_d$ are b-dominating. So (1) holds.

Now consider any vertex x in G_v. Then $x \in N(v_i)$ for some i, $1 \leq i \leq d$. Vertex x has no neighbor in N_i, for otherwise G would contain a cycle of length three. Vertex x cannot have two neighbors $y, z \in N_j$ ($j \neq i$), for otherwise x, y, z, v_j would induce a cycle in G. Then x cannot have two neighbors $y \in N_j$, $z \in N_k$ (with i, j, k pairwise different), for otherwise x, y, z, v_j, v_k, v would induce a cycle in G. Thus x has at most one neighbor in G_v. It follows that the maximum degree in H_v is at most $1 + (d - 2) = d - 1$. Moreover, there cannot be two edges $xy, x'y'$ between N_i and N_j, for otherwise v_i, x, y, x', y', v_j would induce a cycle in G. Thus G_v has at most $d(d-1)/2$ edges. Then H_v does not contain a complete graph K_d, because its largest cliques are induced by the sets $N(v_i) \setminus \{v\}$ and have size $d - 1$. Also, if $d - 1 = 2$, H_v does not contain an odd cycle, because when $d = 3$, the graph H_v has 6 vertices, so the only possible odd cycle is a C_5, and some vertex of the C_5 would have degree 2 in G_v. It follows from Theorem 4 and Claim (1) that $b(G) = d + 1$. □

3 When the degree is small

Theorem 5 *Let G be a d-regular graph with girth $g(G) \geq 5$, different from the Petersen graph, and with $d \leq 5$. Then $b(G) = d + 1$.*

Proof. First let $d = 1$. Then G is a matching and clearly $b(G) = 2 = d + 1$.

Now let $d = 2$. Then G is a disjoint union of cycles, all of length at least 5. Then $b(G) = 3$. Indeed, in order to obtain a b-coloring with 3 colors, it suffices to give colors $1, 2, 3, 1, 2$ to five consecutive vertices in one cycle of G, and to color the rest of G with colors $1, 2, 3$.

Now we assume that $d \geq 3$. We may assume that G contains a cycle of length 5, for otherwise the result follows from Theorem 2. Let $x_1, \ldots, x_5$ be five vertices of G that induce a cycle C in this order. From now on indices are understood modulo 5.

Let $d = 3$. For each $i = 1, \ldots, 5$, let u_i be the neighbor of x_i that is not in the cycle C. First suppose that the edge $u_i u_{i+2}$ exists for every $i = 1, \ldots, 5$. Then the vertices $x_1, \ldots, x_5, u_1, \ldots, u_5$ induce the Petersen graph and form one connected component of G. Since G itself is not the Petersen graph, it must have another component Z. In that case, give color 1 to x_1, x_4, color 2 to x_2, u_5, color 3 to x_3, u_1, and color 4 to x_5, u_2, u_3 and to some vertex z of Z, and give colors $1, 2, 3$ to the neighbors of z. So x_1, x_2, x_3, z are b-dominating vertices of colors respectively $1, 2, 3, 4$, and this coloring can be extended to a coloring of G with 4 colors in any greedy way.

Now we can assume, up to symmetry, that $u_1 u_3$ is not an edge of G. Let us make a b-coloring with 4 colors such that x_1, x_2, x_3, u_2 are b-dominating vertices of colors $1, 2, 3, 4$ respectively. To do this, we first give color 1 to x_1, x_4, color 2 to x_2, color 3 to x_3, x_5, and color 4 to u_1, u_2, u_3. Note that x_1, x_2, x_3 are b-dominating vertices of colors $1, 2, 3$ respectively. Now consider u_2. Let a, b be the two neighbors of u_2 different from x_2. (Possibly $\{a, b\} \cap \{u_4, u_5\} \neq \emptyset$.) Note that a and b are not adjacent to x_1, x_2, x_3, for otherwise G would contain a cycle of length 3 or 4. Moreover, and for the same reason, each of a, b is adjacent to at most one of x_4, x_5; and if each of them is adjacent to one of x_4, x_5 then it is not to the same vertex; in other words the edge set between $\{a, b\}$ and $\{x_4, x_5\}$ is a matching of size at most two. So it is possible to give color 1 to one of a, b and color 3 to the other without having two adjacent vertices of the same color. Now u_2 is a b-dominating vertex of color 4. Finally this coloring can be extended to a coloring of G with 4 colors in any greedy way.

Now let $d = 4$. Let us make a b-coloring of G with 5 colors such that $x_1, \ldots, x_5$ will be b-dominating vertices of colors $1, \ldots, 5$ respectively. To do this, we first give color i to x_i $(i = 1, \ldots, 5)$. For each $i = 1, \ldots, 5$, let u_i and v_i be the two neighbors of x_i that are not in the cycle C, and let $B_i = \{u_i, v_i\}$. We make a few observations. The set B_i is a stable set, for otherwise G would contain a cycle of length 3. For $1 \leq i \neq j \leq 5$, the sets B_i and B_j are disjoint, for otherwise G would contain a cycle of length 4. So $|B_1 \cup \cdots \cup B_5| = 10$. Moreover, for $i = 1, \ldots, 5$, every vertex of B_i has no neighbor in $B_{i-1} \cup B_{i+1}$ and has at most one neighbor in B_{i+2} and at most one neighbor in B_{i-2}, for otherwise G would contain a cycle of length 4. Let H be the subgraph of G induced by $B_1 \cup \cdots \cup B_5$. By the preceding observations, in H every vertex has degree at most 2. First suppose that every vertex of H has degree exactly 2. In this case we call this graph H^*. So H^* must be either a cycle C_{10} or the union of two C_5's.

If H^* is a C_{10}, then, up to symmetry, we may assume that its vertices lie in the following order: $u_1, u_3, u_5, u_2, u_4, v_1, v_3, v_5, v_2, v_4$. If H^* consists of two C_5's, then, up to symmetry,

we may assume that the first C_5 is induced by vertices u_1, u_3, u_5, u_2, u_4 in this order and the second C_5 is induced by vertices v_1, v_3, v_5, v_2, v_4 in this order. In either case, we give colors $3, 1, 2, 4, 1, 4, 5, 3, 5, 2$ respectively to $u_1, u_3, u_5, u_2, u_4, v_1, v_3, v_5, v_2, v_4$. It follows that $x_1, \ldots, x_5$ are b-dominating vertices. This coloring can be extended to a coloring of G with 5 colors in any greedy way.

Finally, if H has vertices of degree less than 2, then, using the above observations, it is a routine matter to check that, up to symmetry, H is a subgraph of H^*, so the same argument can be applied.

Now let $d = 5$. We can assume that G contains a cycle of length 6, for otherwise the result follows from Theorem 3. Let $x_1, \ldots, x_6$ be six vertices of G that induce a cycle C in this order. From now on indices are understood modulo 6. Since G contains no cycle of length 3 or 4, it is easy to see that any vertex of $G \setminus \{x_1, \ldots, x_6\}$ has at most two neighbours in $\{x_1, \ldots, x_6\}$ and if it has two they must be x_h, x_{h+3} for some $h \in \{1, 2, 3\}$. For each $i = 1, \ldots, 6$, let u_i, v_i, w_i be the three neighbors of x_i that are not in the cycle C, and let $B_i = \{u_i, v_i, w_i\}$. Since G contains no cycle of length 3 or 4, it is easy to see that the following observations are true, for all $i = 1, \ldots, 6$:

$$B_i \text{ is a stable set;} \tag{2}$$

$$|B_i \cap B_{i+3}| \leq 1; \tag{3}$$

$$B_i \cap B_{i+1} = B_i \cap B_{i+2} = \emptyset; \tag{4}$$

$$\text{There is no edge between } B_i \text{ and } B_{i+1}; \tag{5}$$

$$\text{For } j \neq i, \text{ the edges between } B_i \text{ and } B_j \text{ form a matching.} \tag{6}$$

In view of (3), we distinguish between two cases.

Case 1: $B_1 \cap B_4 = B_2 \cap B_5 = B_3 \cap B_6 = \emptyset$.
So the six sets $B_1, \ldots, B_6$ are pairwise disjoint, and $|B_1 \cup \cdots \cup B_6| = 18$. Let us make a b-coloring of G with 6 colors such that $x_1, \ldots, x_6$ will be b-dominating vertices of colors $1, \ldots, 6$ respectively. To do this, we first give color i to x_i $(i = 1, \ldots, 6)$. In order to make x_i a b-dominating vertex, we want to give colors $i+2, i+3, i+4$ to the three vertices of B_i. Let us show that such a coloring exists. Since the set $S_i = \{i + 2, i + 3, i + 4\}$ of colors we want to give to the vertices of B_i is disjoint from the set S_{i+3}, the edges between B_i and B_{i+3} are immaterial. Therefore let H be the subgraph of G induced by $B_1 \cup \cdots \cup B_6$ and excluding the edges between B_i and B_{i+3} for all $i = 1, 2, 3$. By the above observations and by the definition of H, there is no edge of H between $B_1 \cup B_3 \cup B_5$ and $B_2 \cup B_4 \cup B_6$. Let J be the subgraph of H induced by $B_1 \cup B_3 \cup B_5$. By the observations, every vertex has degree at most 2 in J, and any cycle in J has length a multiple of 3. We may consider three special cases for J. Let J_9 be the 9-cycle $u_1, u_3, u_5, v_1, v_3, v_5, w_1, w_3, w_5$ in this order. Let J_6 be the graph in which $u_1, u_3, u_5, v_1, v_3, v_5$ induce a 6-cycle in this order and w_1, w_3, w_5 induce at most two edges. Let J_3 be the graph whose edges are $u_1u_3, u_3u_5, v_3v_5, v_5v_1, w_5w_1, w_1w_3$. We claim that:

$$\text{Up to symmetry, } J \text{ is a subgraph of one of } J_9,\ J_6,\ J_3. \tag{7}$$

We only sketch the proof of this claim. If J contains a cycle, then this cycle has length 6 or 9 because G contains no 3-cycle, and so, up to symmetry, either $J = J_9$ or J is a subgraph

of J_6. If J has no cycle, then there is a vertex, say w_5, such that none of u_3w_5, v_3w_5, w_3w_5 is an edge of J; and then, if J is not (up to symmetry) equal to J_3, it is possible to add one of these three edges to J in such a way that (2)–(6) still hold and without creating any cycle of length 3 or 4. Thus the claim follows by induction on the value of $9 - |E(J)|$. So the claim holds.

Now we give colors $3, 5, 3, 4, 6, 1, 5, 1, 2$ to $u_1, u_3, u_5, v_1, v_3, v_5, w_1, w_3, w_5$. In either of the three possibilities mentioned in the claim, this is a proper coloring of J. Then x_1, x_3, x_5 become b-dominating vertices of colors $1, 3, 5$ respectively. The same argument can be applied to the subgraph of H induced by $B_2 \cup B_4 \cup B_6$, so that x_2, x_4, x_6 become b-dominating vertices of colors $2, 4, 6$ respectively. Finally this coloring can be extended to a coloring of G with 6 colors in any greedy way.

Case 2: $B_1 \cap B_4 \neq \emptyset$.
Let $y = w_1 = w_4$ be the unique vertex in $B_1 \cap B_4$. We are not assuming that $B_3 \cap B_6$ is empty. We will make a b-coloring of G with 6 colors such that $x_1, \ldots, x_5$ are b-dominating vertices of colors $1, \ldots, 5$ respectively, and y is a b-dominating vertex of color 6. To do this, we first give color i to x_i $(i = 1, \ldots, 5)$ and color 6 to y. Also we give color 3 to x_6. Let $B_y = N(y) \setminus \{x_1, x_4\}$. Since G contains no cycle of length 3 or 4, we observe easily that B_y is a stable set, B_y is disjoint from $\{x_2, x_3, x_5, x_6\} \cup B_1 \cup \cdots \cup B_5$, there is no edge between B_y and $B_1 \cup B_4$, and for $j = 1, \ldots, 6$, the edges between B_y and B_j form a matching. For $i = 1, 4$, let $B'_i = B_i \setminus \{y\}$.

First suppose that $B_2 \cap B_5 = \emptyset$. In order to make x_1 a b-vertex, we must assign colors $4, 5$ to the two vertices of B'_1, which we denote by $B'_1 \to \{4, 5\}$. Likewise we want $B_2 \to \{4, 5, 6\}$, $B_3 \to \{1, 5, 6\}$, $B'_4 \to \{1, 2\}$, $B_5 \to \{1, 2, 6\}$, and $B_y \to \{2, 3, 5\}$. Let us show that such a coloring exists. Consider the subgraph J of G induced by $B_2 \cup B_5 \cup B_y$. This subgraph, together with the coloring requirements for B_2, B_5, B_y, is isomorphic to the graph "J" defined in Case 1, and so, by the same arguments, it can be colored as required. Then we color the vertices of B_3 as follows. Since there is no edge between B_3 and B_2, and each vertex of $B_5 \cup B_y$ has at most one neighbour in B_3, and x_6 has the same color as x_3, we have the constraint that at most one vertex of B_3 cannot have color 5 and at most one vertex of B_3 (possibly the same) cannot have color 6, and no other constraint. So, in either case, it is easy to see that the three vertices of B_3 can be colored $1, 5, 6$ in a way that respects these constraints. Then, we color B'_1 as follows. Since there is no edge between B'_1 and $B_2 \cup B_y$, and the vertices of B_6 are not colored with 4 or 5, and every vertex of B_3 has at most one neighbour in B_1, we have the only constraint that at most one vertex of B'_1 cannot have color 5. So the two vertices of B'_1 can be colored $4, 5$ in a way that respects this constraint. Then we can assign colors $1, 2$ to the vertices of B'_4 arbitrarily, because there is no edge between B'_4 and $B_3 \cup B_5 \cup B_y$ and the vertices of $B'_1 \cup B_2$ have colors different from $1, 2$. Finally this coloring can be extended to a coloring of G with 6 colors in any greedy way.

Now suppose that $B_2 \cap B_5 \neq \emptyset$, and let $z = w_2 = w_5$ be the unique vertex in this intersection. Assign color 6 to z. For $i = 2, 5$, let $B'_i = B_i \setminus \{z\}$. Now we want $B'_2 \to \{4, 5\}$, $B'_5 \to \{1, 2\}$, while the coloring requirements for B'_1, B_3, B'_4 and B_y remain the same. Let J' be the subgraph of G induced by $B_3 \cup B_5 \cup B_y$. This subgraph, together with the coloring

requirements for B_3, B_5, B_y, is isomorphic to an 9-vertex subgraph of the graph "J" in Case 1, and so, by the same arguments, it can be colored as required. Then we color B'_2 as follows. Since there is no edge between B'_2 and B_3, and the vertices of B_5 are not colored with 4 or 5, and every vertex of B_y has at most one neighbour in B_3, we have the only constraint that at most one vertex of B'_2 cannot have color 5. So the two vertices of B'_2 can be colored $4, 5$ in a way that respects this constraint. Then we can color the two vertices of B'_1 with colors $4, 5$, the only constraint being that at most one of them cannot have color 5 (because of some edge between B_3 and B'_1). Then we can color B'_4 arbitrarily with $1, 2$ as in the preceding paragraph. Finally this coloring can be extended to a coloring of G with 6 colors in any greedy way.

□

Our findings lead us to a new formulation of El-Sahili and Kouider's question:

Conjecture 1 *Every d-regular graph with girth at least 5, different from the Petersen graph, has a b-coloring with $d + 1$ colors.*

References

[1] C. Berge. *Graphs.* North Holland, 1985.

[2] A. El-Sahili, M. Kouider. About b-colourings of regular graphs. Res. Rep. 1432, LRI, Univ. Orsay, France, 2006.

[3] T. Faik. La b-continuité des b-colorations: complexité, propriétés structurelles et algorithmes. PhD thesis, Univ. Orsay, France, 2005.

[4] S. Gravier. Coloration et produits of graphes. PhD thesis, Univ. Grenoble, France, 1996.

[5] F. Harary, S. Hedetniemi, G. Prins. An interpolation theorem for graphical homomorphisms. *Port. Math* 26 (1967) 453–462.

[6] M. Kouider. b-chromatic number of a graph, subgraphs and degrees, Res. Rep. 1392, LRI, Univ. Orsay, France, 2004.

[7] R.W. Irving, D.F. Manlove. The b-chromatic number of graphs. *Discrete Appl. Math.* 91 (1999) 127–141.

[8] D.F. Manlove. Minimaximal and maximinimal optimisation problems: a partial order-based approach. PhD thesis. Tech. Rep. 27, Comp. Sci. Dept., Univ. Glasgow, Scotland, 1998.

Migrating Web Application towards Web Services Technologies

Mostefai Aek [1],Malki Mimoun [2] ,Bouchiha Djelloul [3]

{1.2.3} EEDIS Laboratory, Djilali liabess university, Sidi belabess, Algeria.
[1]Mostefaia_aek@yahoo.fr.,[2] mymalki@gmail.com, [3] Bouch@yahoo.fr

Abstract. There are large demands from organizations to migrate human oriented Web application (WA) systems to web services technologies, a solution is the use of process , methods to reengineer WA to Web services , this solution eliminate the necessity to redevelop this systems from zero . In This paper we present a process, WA2WS, and its related tool to migrate WA towards Web services technologies, the process is composed of seven phases including activities of reverse engineering , and forward engineering ,the reverse engineering activities are task domain ontology based , on aiming at the discovery of functionalities to be exposed as web services ,UML state transition diagram (STD) describing service navigation path , and needs information to export WA as web services . The forward engineering activities are centered on the construction of the wrapper that acts like an interpreter of the inferred STDs.

Keywords: Migration, reverse engineering, Task domain ontology, reengineering, Wrapper, STD, Web Application, Web Services, WA2WS.

1. Introduction

An ever growing number of applications can be accessed through the World Wide Web these applications are referred to web applications. However, those applications have a major drawback – they are intended for manual use only. Applications are operated by means browsers. Moreover, integrated information processing across the boundaries of individual Web resources is a difficult problem using the existing Web infrastructure of today.

The SOA (i.e., Service Oriented Architecture) is based on an idea that application can be developed by composing existent services offered by different providers and the services web define the technology of this idea .

Shirky [1] suggests that web services are focused on application program interoperability (to aim to repeat the success achieved by the web in document access), thus web services do not need to be invoked by a browser, but by other application programs. The most general form involves binding complex programs together from component parts anywhere in the world. General inter-operability has been tried before, but with partial success, for example DCOM, Corba, and RMI. With these, both the client and server have to load the system; with web services, the idea is to know

nothing about the "other end" other than what can be communicated via standard protocols.

For this, WSDL [3] (Web Services Description Language) is used to define how the components are specified .How are published to be discovered and reused through an UDDI [4] (i.e., Universal Description, Discovery and Integration) and how are invoked at run time with SOAP protocol [5] (Simple Object Access Protocol). This solution technology raise large demands from organizations to migrate human oriented Web application systems to web services technologies, where the use of reengineering technique eliminate the necessity to redevelop this systems from zero.

The migration of applications to the context of service provider can be approached by a white box modernization technique which needs an understanding, and a modification of source code of application, by black box technique that needs the comprehension of interface and the interactions of user with application and without modification of code, or by grey box technique which mixes the two approaches [6].

Adding a software layer called often wrapper, able to expose the new functionalities (i.e., web services) of the web application in architecture SOA, and to resolve the mismatches between the old and the new interface are black box modernization technique. To identify functionalities to be exported towards web services technologies some factors should be resolved, these factors are properly related to characteristics of web application :

1. The service discovery including the identification of inputs/outputs, where the major approaches take into account service outputs.
2. Generation of extraction motifs for the data service output, or for the intermediately needed data to continue the execution of web application taking into account that this data are encapsulated in ill formed HTML documents.
3. Generation of mapping rules to encapsulate extracted data into XML documents, necessary for building SOAP response messages.
4. Building of the service navigation path, when service escape from request/response model in web application, or when service output data are broken in more than one HTML page.

The proposed approach gives answers to all this preoccupation, and grave into black box modernization technique, where the Wrapper is the key component of the migration process, black box here means that we do not need to understand, and modify the source code of web application scripts.

The paper is organized as follow, Section 2 presents the related works on the reengineering of web applications toward the web services technologies, Section 3 discuss the WA2WS, the proposed process, section 4 presents the experimentation and the used issues in the development of WA2WS tool, a conclusion and future orientations are presented in the Section 5.

2. Related works

Many approaches had been proposed in the literature for the reengineering of the web applications and the extraction of information from its content .In [7] they presented an approach to WWW information integration, based on the development of a canonical domain model in XML and the wrapping of existing WWW applications with wrappers capable of communicating about entities in this common model with the applications and with an intermediary mediator , [8]propose a static web sites re-

engineering approach toward the grid oriented services with the creation of state full resource,[9] they have chosen to do the retro engineering of the "layer Presentation" of the web application to extract from its content set of functionalities actually supplied , in [10] an approach to build web services from CGI web sites using a semi automatic method of Wrapper generation, In [11] integration consist to exploit the interfaces of web applications, and to convert the answers HTML documents in XML documents .To permitting the integration, they had developed a set of tools which allow the development of applications and the implementation of the process without necessity to manual coding . Lixto [15] is a tool for wrapper generation, very well adapted for the construction of wrappers HTML / XML.Web applications integration can achieve by some dedicated wrappers , Wrappers that can extract data from web application interface , for more details see [12] ,[13] propose an approach based on the separation of treatment from data base ,to migrate the legacy e-commerce web applications toward the EJB architecture.
The problem of migration of web application from a framework to another for example ASP towards NSP is addressed [14] using a method called water transformation. Tilley and others describe the inherent challenge in the migration to web services [16].

3. The proposed process

The WA2WS (*Web Application To Web Services*) migration process, see Figure 1, allowing the functionalities implemented by WA to be accessed as a web service ,is task domain ontology [18] based ,see Figure 2, need that we start the execution of WA, in some case more than one execution , and includes the seven phases where two can be conduced in parallel ,
The first four phases (i.e., service discovery, tracking user's actions, navigation model building, and service description generation) are activities of reverse engineering process, aiming at:
Discovering functionalities to be exposed as web services in terms where inputs and outputs services are identified, the identity of service navigation path modeled by UML state transition diagram, and the generation of declarative service description, these description are the transformation of the UML state transition diagram, it is important to say that the extraction motifs, mapping rules are identified and generated at this stadium.
The rest of phases are forward engineering activities, centered on the construction of the wrapper, where the mission is the execution of web application following the services descriptions, it acts like an interpreter of the UML state diagrams. All this phases are described below.

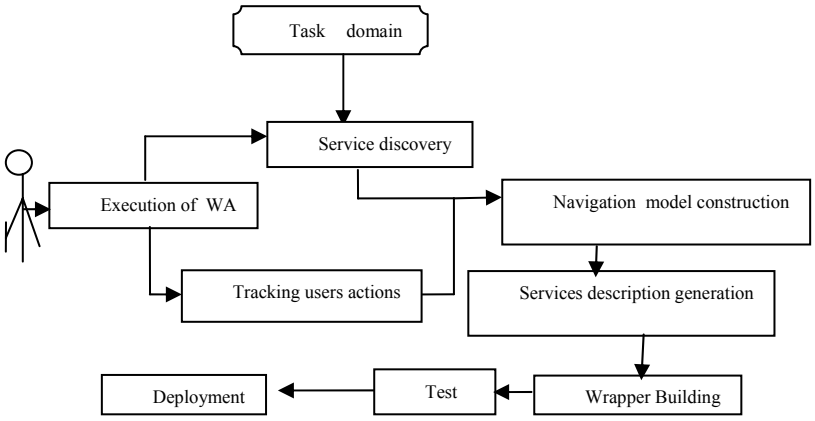


Fig. 1. The suggested WA2WS migration process

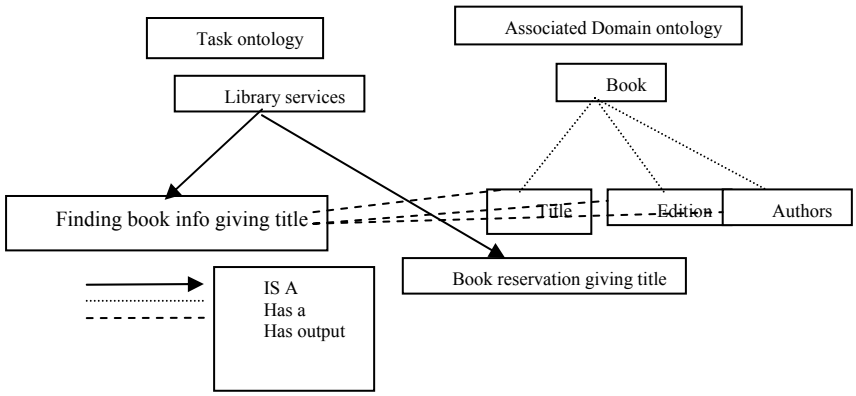


Fig. 2. Task domain ontology and domain ontology (partial).

3.1 Services Discovery

The web application offers services to end users by browsers, the entry to these services are generally the web forms which expose a set of components (i.e., radio button, check box, choices list, zone of seizure, field of seizure) these forms are identified by the opened markup <form >and the closed one </form>,and generally, web pages of web application containing the results are formatted using some structured forms of displays (i.e., table, lists), HTML forms are analyzed to capture the candidate services inputs, and response web pages of WA interrogation request are processed to capture the candidate service outputs, our idea behind this that WA are labeled where meaningful information are placed side issue a component in HTML form , or near outputs data , this information give users an understanding

signification about data formatted in web pages (e.g., Name : , Title : ,).The task domain ontology describe in an abstract way the set of tasks provided in this domain , this tasks are specified in terms of inputs/outputs and implemented in OWL [17] , WA offers services in specific domain, so services to be discovered are one , an aggregation , of the ported in task ontology, in this sense task ontology gives a general view about task to be found in a specific domain.

It is very important to say in this phase; all needed information for extraction (i.e., motifs rules, or Xpath rules), and for mapping are identified, and generated, how we proceed, details are given in the following sections

3.1.1 Identification of candidate inputs and candidate outputs

Labels presents in HTML forms are identified, and extracted, here all labels that are present near a component (e.g., Text field, radio button …) , In some cases not all extracted labels can be an input of service, one's can be a parameter where its value is fixed in all execution (e.g., radio button : borrowed book) ,this fixed inputs (i.e., inputs with fixed value), are identified in second execution of the WA , text fields are mandatory inputs , radio button can take a role of parameter input when its value are fixed . To identify candidate outputs we follow the same principal in the identification of candidate inputs, where, here all labels presented in web pages result of querying WA are identified, and extracted.

3.1.2 Services identification

An algorithm, See Figure 3, had been developed to identify the functionalities to be exported as web services, this algorithm take a task domain ontology, and set of candidate inputs ,outputs and tries identifying set of task that match better the Is/Os identified from WA , the algorithm works at run time, to decide when two terms are matched ,we need to define similarity measure ,we use two measure :

Syntactic similarity

[21] Introduce a measure to compare two strings of characters called the edition distance, in our purpose to calculate similarity we use the measure of syntactic similarity introduced in [22], that is the inverse the edition distance (Ed).

$$\text{Sim syntactique } (v1, v2) := \max\left(0, \frac{\min(|v1|,|v2|) - ed(v1,v2)}{\min(|v1|,|v2|)}\right) \quad (1)$$

The idea that is ,to make two string of characters and to calculate the number of the necessary actions to transform a one string to the other one ,the atomic actions are insertion ,suppression ,modification and also the displacement of characters.

Semantic similarity

Make references to the similarity between two concepts in taxonomy [23] as WORDNET, different approach has been proposed to calculate the similarity between two concepts of an ontology for example WORDNET, [25] presents a comparisons between the proposed methods , where the best is measure of wu-palmer [24]. That we use here. In domain of concepts, similarity is defined as the reports between the distance which separates the two concepts in the hierarchy and by the report of their position to the root.

Similarity between C 1 and C2 is:

$$\text{ConSim } (C1,C2) = \frac{2*N3}{N1+N2+2*N3} \quad (2)$$

Formally this measure will be:

$$ConSim(C1, C2)=(2*depth(C)/(depthc(C1)+depthc(C2)) \quad (3)$$

Where C (i.e., Concept) is the LCS (i.e., Lower Common Subsumer) of C1 and C2 (in number of edges), depth(c) is the number of edges who separates Ci of the root passing by C, this measure has the advantage to be simple and to have best performances [25].

Inputs : task domin ontology , SDI(set of discovered inputs) ,SDO(set of discovered outputs).
Outputs : SSD (set of discovered services , SND (set of not matched inputs)
SNOD (set of not matched outputs).
Begin
SND={} ;SME (set of marked inputs)= {} .
/*marking inputs */
buildd the set SOI of inputs tasks in task ontology .
For each input e in SDI do
For each input eo in SOI do
$\alpha = match_{SYN}(e, eo)$;
if $\alpha>\theta$ then SME=SME+e
else { $\alpha=match_{SEM}(e,eo)$
if $\alpha>\delta$ then SME=SME+eo
else SND=SND+eo}
/*marking outputs */
SMO (set of marked outputs)= {}.;SNOD (set of not marked outputs)= {}.
Build the set SOO off all outputs task in task ontology ..
For each output S in SDO do
For each output so in SOO do
$\alpha = match_{SYN}(S, So)$;
if $\alpha>\theta$ then SMO=SMO+So
else { $\alpha=match_{SEM}(S, So)$
if $\alpha>\delta$ then SMO=SMO+So
else SOND=SOND+So}
SDS (set of discovered services) ={} .
For each task in task ontology do {
Build the set SET of its inputs ;
Build the set SOT of its outputs ;
If (SET $\subset$ SME) and (SOT $\subset$ SMO) then
SDS =SDS+{task name } ;
Return SDS ,SND, SOND;
End.

Fig. 3. Services discovery Algorithm

3.2 Tracking and recording user actions

The user who exploit a service of web application using its browser generally make several actions until the desired result is viewed ,for example to consulate the received e-mails in the portal YAHOO at the address www.yahoo.fr , the user tape the URL of the site ,log in by filling the form "LOGIN" and click in the button login ,next click in the link " boite de reception ", in the most of web application (e.g., e-commerce web application) session keys , and cookies are generated automatically , in a automated access to the WA ,security mechanisms obligate programs to start from index pages where session keys are generated, this aspect must be taken into account . To automate the access to services of the WA (i.e., building web services),we

have to record for each services the necessary actions to be performed on the WA later in response to the demand request of web service ,this necessary actions can be classified into three actions: URL input, click (radio / select option / button/link) and data input . These "actions" are the activities of user's navigation in WWW and need to be recorded and serve as the basis for inferring the service UML state transition diagram.

3.3 Navigation model construction

For each identified web service, a path of navigation in the WA is determined; the path of navigation includes the starting URL, the intermediate actions and the end web pages of the service. We propose UML state transition diagram to model the path of navigation instead of navigation maps [26] ,the idea behind the use of UML state diagram (i.e.,STD) is to represent the actions performed on the WA specially where we should extract the result data starting from the observation that in WA the result data are displayed in many/different linked web page(e.g., the simplest is the presence of the link next).

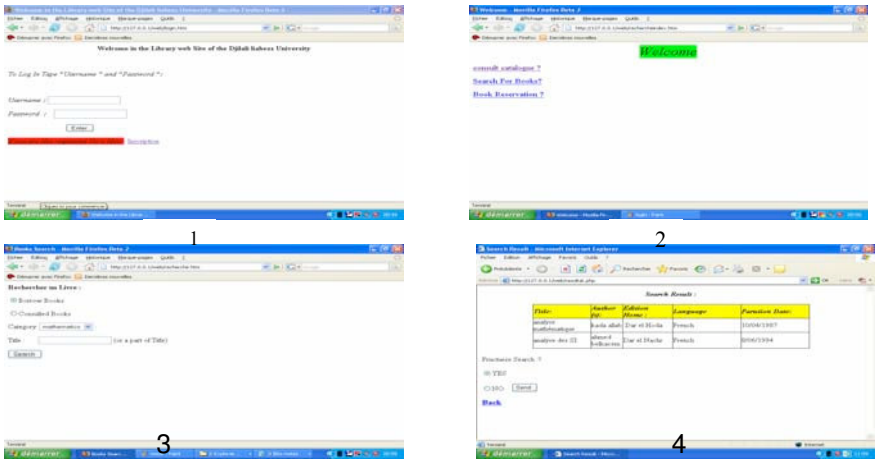
The UML state diagram is inferred by a process of reverse engineering applied on the data recorded by the phase “collect the track of execution “. Each UML STD represents a slice of WA from service view point.

3.3.1 Service navigation modeling in WA

Web pages represent the active part in the system, elements in web page (e.g.,link), Change the system when click occur in it ,so :

1. Web pages are represented by state in the diagram.
2. Link clicks are events occurring in WA changing the state of the system are represented by transitions.

the event link click is denoted by the symbol AL (link name), and button click leading to action performed in button to submit a form after filling it is denoted by the symbol SForm(parameters) ,parameters stand to inputs filled in the form and sended to the WA . Figure 4, presents a service path navigation and its inferred UML state transition diagram , marked with where we should extract the response data; here we present a demonstrative example that have major characteristics of web applications .



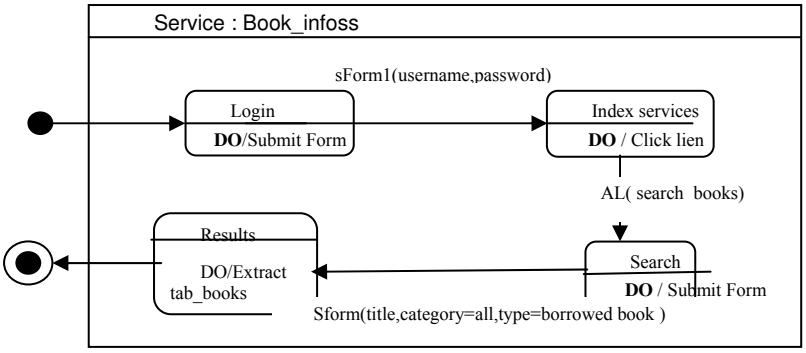


Fig. 4. Service navigation path and its inferred STD.

3.4 Generating service description

The inferred service UML STD, gives an abstract view of how to navigate in WA to have the desired data.

This model needs to be traduced in some declarative description to be executed by deduced interpreter called a wrapper. Starting from this, we need to model WA objects, in the literature several data models to represent the Web have been proposed (e.g., [26, 27, 28]). Site descriptions based on these data models provide a uniform representation of sites that is similar to a schema in traditional databases, and thus simplify querying of described sources. They are also focused, and need only reflect a set of pages that contain relevant information and services. Since these models contain not only information about the contents of pages, but also information required to navigate through a set of pages. Service description used in our approach are the extension of navigation map described in [29], with this description we can handle all the necessary information needed for navigating, retrieving and extracting data from WA . The service UML state transition diagram describing the process of navigation in the WA is modeled on XML between collections of objects F-logic [30]. That

represents a slice of WA from service viewpoint. Due to the limited space we do not give the details on how transform the UML state transition diagram to services description. As an example, the objects of our web site are depicted in Figure 5.

```
Libray:WebPage[
Adress→'http://www.localhost:8080/web/Login.html',
Title->'bibliotheque',
Action->{a1}]
A1 :action [Form->f1[script←'login.php3',
Method->'post',
Mandatory->> {attr1, attr2}]
Source->'http://www.localhost:8080/Login.html',
Destination->p2]
p2 :WebPage[ adresse ->'http://www.localhost:8080/web/index',
……
p4 :WebPage[adresse->'http://www.localhost:8080/web/resultat.php'
Titre->'result'
Extracted_attr->result_tab]
Attr1:simple_attribute[
Type→string
Nom→'username'
Attri_frm_name→user]
……
Xpath->//html/body/table[0]
Fields->>{edition,author,language}]
……..
xpath->//html/body/table[0]/tr[x]/td[2]]
language:atomic_attri[
xpath->//html/body/table[0]/tr[x]/td[2]]
```

Fig. 5. Web Objects of WA in F-logic (Partial).

We use XML to model the process of navigation, generally, this define an ordered way to exercise actions on WA to have the desired result, so the semantic of the used Tags are intuitively understand for example the semantic of <follow_link >a2 <follow_link > is : click on the link of action a2 , the necessary information needed to perform action are inferred from f-logic database . Figure 6 give an example of the books_info process result of translation of the STD of Figure 4.

```
<service>
<name>Books_info</name>
< begin_ url>http://www.localhost:8080/web/login.html</ begin_ url >
<submit_form>a1</ submit_form >
<follow_link >a2< follow_link >
< submit_form >a3</ submit_form >
<Extract>result_tab</extract>
</service>
```

Fig. 6. Service description of: books_info service.

3.5 Wrapper building

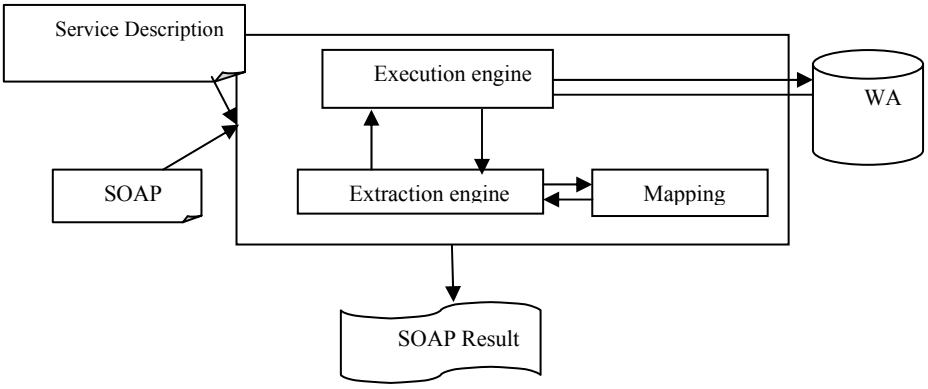


Fig. 7. Wrapper architecture.

The goal of the wrapper is to take service description and the request SOAP , execute the WA ,extract needed data and derive the SOAP response .the main objective is to build generic wrapper in the sense that it would be able to execute any new description of any WA , so it can be reused in migrating any other WA .another important reason behind the generic solution is to deal with the presentation change in WA , so we do not need to rebuild the service but to reuse the WA2WS systems to infer the new description of changed service and the wrapper will be able to respond correctly to the requested service , the main architecture of the wrapper is depicted in Figure 13 , generally speaking the wrapper act as an interpreter of services description . Wrapper principle components are, execution engine extraction engine and the mapping engine , all these components interact together in order to respond to a demand SOAP message.

3.5.1 Execution engine

This module has the responsibility to execute the WA following service description, load page follow link, and submit form, handle cookies and session are the main principal functions of this component. The following algorithm (Figure 8) express how to execute the WA based on the service description generated previously.

```
Inputs: XML file of service description, SOAP demand
Outputs: WA executed.
Begin
Set inputs values in F-logic database base on the received value on the SOAP message.
Load XML file of service description
Repeat
If markup=submit_form then {
        Infer the needed data from the F-logic database;
            Call Submitform ();     } Else  If markup=follow_link then {
    infer the needed data from the F-logic database;
        Call Clicklinck();    } Else If markup=extract_data then {
                            Call extraction ();}
Until no action found ;  End.
```

Fig. 8. Execution Algorithm

3.5.2 Extraction engine

To build the SOAP response we should extract data from the web page of the application. This data have to be extracted from many web page during the execution, in the simple case from the latest web page loaded, in other, some extracted data are necessary to the continuous of the execution of the WA (e.g., link dynamically generated). How we extract? the main principal technique used to extract web data is to use pattern of extraction (i.e., relative Xpath expressions) applied on defined web page .the extracted are rendered to the execution engine in order to continue the execution ,or to the mapper to build the SOAP message , our idea behind the separation of the extraction engine and the execution engine is to give a large free to use any available Wrapper (here the wrapper stands for program that extracts data from web page) , wrapper ontology based can easily inserted in our wrapper for more about approaches to build these programs see [12] . We have adopted a DOM[31] representation ontology based to build this program, taking into account the advantage of relative Xpath expression to build a robust wrapper.

Relative Xpath expression identification

When a co (i.e., candidate output) are identified in the service discovery phase ,we build the DOM document where co appear then we generate a relative Xpath expression identifying the co. to clarify Figure 13 give an example of a DOM document containing COs .

So, the relative Xpath expression //html/body/table[0]/tr[x]/td[2] are generated to extract all books edition home, we have used the algorithm presented in [35] to infer this expression .

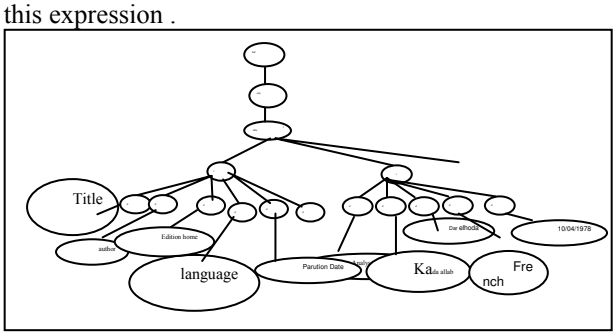


Fig. 9. DOM tree of result web page.

3.5.3 The mapping engine

//html/body/table[0]/tr[1]/td[2] extract some data from web page (e.g., dar el hoda) but we need to determine the semantic of this data ,dar el hoda stand for edition home , in the SOAP message we have to see <edition> dar el hoda </edition>. The goal of the Mapper is to annotate the extracted data; the final task of it is the result XML document. Annotating, is not an easy task, [35] state a solution ,but intervention of user is needed to map the web page outputs to an Xml schema (i.e., annotation manually) .

Our solution is to fully automate the process given in [35], the task ontology accompanied with the domain ontology simplifies the process, in this sense the XML

schema is inferred from the outputs of the service matched by some task in the task ontology. Hence, when CO are identified in the discovery phase, a mapping rule is generated, the cardinality in the task ontology tell about the repetition number of the elements, due to the limited space we do not give a detail on how to infer XML schema from task ontology .

3.6. Test and deployment

In order to assure that the constructed service work correctly, we have to use different inputs to call the service, the same inputs have to be used in the execution of WA, and the two results are compared pair to pair.

After confirmation that all works, WSDL description are generated, this is a sample task we need to deploy the Wrapper with service description on a server (e.g., apache axis) or the use of java2wsdl program. This description can be exported to an UDDI.

4. Experimentation

To validate the effectiveness of our approach we have developed a tool named WA2WS tacking his name from the name of the process. The tool is 100 % java.The tool supports the first four phases of the process (i.e., until the generation of the services description).the wrapper is also implemented on java,to tracking the users actions Mozilla Gecko [33] and Jrex [34] are used to personalize our browser , Jena API is used to manipulate the task ontology .HTML document are ill formed so HTMLparser is used to extracted the needed data from web page in the discovery phase .service description are generated according to Flora-2 [36]program. For the wrapper execution engine is implemented basing on Httpunit, extraction engine on DOM4J.

4.1 Case study

WA2WS tool and our wrapper is used to migrate the WA of university library developed in PHP. Figure 12, illustrates the WSDL document of the service "books_info" (Livres_info). The test phase had been past successfully.

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions targetNamespace="http://livre_info/" xmlns:apachesoap="http://xml.apache.org/xml-soap"
xmlns:impl="http://livre_info/" xmlns:intf="http://livre_info/" xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/" xmlns:wsdlsoap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
<!--WSDL created by Apache Axis version: 1.2RC2
Built on Dec 08, 2006 (12:13:10 PST) -->
<wsdl:message name="livre_infoResponse">
<wsdl:part name="livre_infoReturn" type="apachesoap:Document"/>
</wsdl:message>
<wsdl:message name="livre_infoRequest">
</wsdl:message>
<wsdl:portType name="livre_info">
<wsdl:operation name="livre_info">
<wsdl:input message="impl:livre_infoRequest" name="livre_infoRequest"/>
<wsdl:output message="impl:livre_infoResponse" name="livre_infoResponse"/>
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="livre_infoSoapBinding" type="impl:livre_info">
<wsdlsoap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
<wsdl:operation name="livre_info">
<wsdlsoap:operation soapAction=""/>
<wsdl:input name="livre_infoRequest">
<wsdlsoap:body encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" namespace="http://livre_info/"
use="encoded"/>
</wsdl:input>
```

Fig. 10. WSDL document of service: books_info.

Other WAs have been migrated successfully with our tool and Wrapper.

5. Conclusions and future works

In this paper we presented the process, WA2WS, and its related tool to migrate WA towards web services technologies and SOA, our tool and wrapper can easily used in migrating WAs with no prior knowledge (e.g., programming languages, technologies knowledge like XML, HTML…), the tool and wrapper are experimented and are successfully used to migrate some WA.

In the future we plan to adapt the process, tool, and Wrapper migrate legacy mainframe and desktop applications, other objective are: evolve the process to compose web applications, and to take into account the not labeled WAs.

6. References

1. A. Arsanjani, B. Hailpern, J. Martin and P. Tarr, "Web Services: Promises and Compromises," ACM Queue, vol. 1, no. 1, pp. 48-58, 2003.
2. Shirky C. Web services and context horizons. IEEE Computer, September 2002, page 98 – 100, vol.35, no. 9.
3. Web Services Description Language (WSDL) : http://www.w3.org/TR/wsdl
4. UDDI technical paper, http://www.uddi.org/pubs/Iru_UDDI_Technical_ White_Paper.pdf
5. Simple Object Access Protocol (SOAP) http://www.w3.org/TR/2003/REC-soap12-part0-20030624/
6. Z. Zhang and H. Yang, "One-Stone-Two-Birds: LegacySystem Re-engineering and Web Services Development - - A Component-Based and Service-Oriented Approach" in Proceedings of Postgraduate Research Conference in Electronics, Photonics, Communications & Networks, and Computing Science (PREP 2004), Hatfield, UK, 2004.
7. Eleni Stroulia, Judi Thomson, Gina Situ. " Constructing XML-speaking Wrappers for WEB Applications: Towards an Interoperating WEB". In the Proceedings of the 7 th Working Conference on Reverse Engineering, Brisbane, Queensland, Australia, 59-68, IEEE Computer Society Press. 23-25 November 2000.
8. Jianzhi Li and Hongji Yang. Towards Evolving Web Sites into Grid Services Environment. Proceedings of the Seventh IEEE International Symposium on Web Site Evolution (WSE'05). 2005.
9. Yingtao Jiang, Eleni Stroulia. Towards Reengineering Web Sites to Web-services Providers. In Proceedings of the Eighth European Conference on Software Maintenance and Reengineering (CSMR'04), pp. 296-305. 2004.
10. Hoang PHAM HUY, Takahiro KAWAMURA, Tetsuo HASEGAWA. "Web service gateway - A step forward to e-business". In Proceedings of the 2004 IEEE International Conference on Web Services (ICWS"04), 2004.
11. Robert B ,Georg G ,Marcus H,Wolfgang S "Interactively Adding Web Service Interfaces to Existing Web Applications" in Proceedings of the 2004 International Symposium on Applications and the Internet (SAINT '04),p:1-7
12. Mostefai .A,Bouchiha .D,Khater M . "génération et maintenance des Wrappers: état de l'art" . CIIA06 SAIDA .pp1-13
13. Lau, T., Lu, J., Mylopoulos, J. & Kontogiannis, K. "The Migration of Multi-tier Ecommerce Applications to an Enterprise Java Environment". Information Systems Frontier 5:2, 149-160. 2003
14. Ahmed E. Hassan and Richard C. Holt : "MigratingWeb Frameworks Using Water Transformations". Proceedings of the 27th Annual International Computer Software and Applications Conference (COMPSAC'03) © IEEE. 2003.
15. R, Baumgartner, S. Flesca and G, Gottlob, "Visual Web information extraction with Lixto," In Proceedings of the 27th International Conference on Very Large Data Bases(VLDB'01), 2001, pp. 119-128.

16. S. Tilley, J. Gerdes, T. Hamilton, S. Huang, H. Muller, K.Wong, "Adoption challenges in migrating to web services", Proceedings of the 4th International Workshop on Web Site Evolution, WSE 2002, IEEE CS Press, Los Alamitos, CA, 2002, pp.:21 – 29
17. OWL (Ontology Web Language):http://www.w3c.org/OWL.
18. Mizoguchi R., Kozaki K., Sano T. et Kitamura Y. (2000). "Construction and Deployment of a Plant Ontology ". The 12th International Conference, EKAW2000, (Lecture Notes in Artificial Intelligence (1937), 113-128
19. Wu Z. & Palmer M. (1994) Verb Semantics and Lexical Selection, Proceedings of the 32nd Annual Meetings of the Associations for Computational Linguistics, pages 133-138.
20. Karl R.P.H. Leung_ Lucas C.K. Hui_ S.M. Yiu_ Ricky W.M. Tang ModellingWeb Navigation by Statechart HKU CSIS Tech Report TR-2000-01
21. Levenstein, I.V.: Binary codes capable of correcting deletions, insertions, and reversals. Cybernetics and Control Theory (1966)
22. Maedche, A., Staab, S.: Measuring similarity between ontologies. In: Proceedings of the European Conference on Knowledge Acquisition and Management (EKAW), Springer LNCS (2002)
23. Resnik, P. (1995b). Using information content to evaluate semantic similarity in a taxonomy. In Proceedings of IJCAI-95, pages 448–453, Montreal, Canada.
24. Fellbaum C. (1998) WORDNET. An Electronic Lexical Database. In The MIT Press.
25. D. Lin. (1998) An information-theoretic definition of similarity. In Proceedings of 15th International Conference On Machine Learning, 1998.
26. H. Davulcu, J. Freire, M. Kifer, and I. Ramakrishnan. A layered architecture for querying dynamic web content. In Proc. of SIGMOD, pages 491–502, 1999
27. P. Atzeni, G. Mecca, and P. Merialdo. To weave the web. In Proc. of VLDB, pages 206–215, 1997.
28. C. Knoblock, S. Minton, J. Ambite, N. Ashish, P. Modi,I. Muslea, A. Philpot, and S. Tejada. Modeling web sources for information integration. In Proc. of AAAI, 1998.
29. Vinod Anupam Yuri Breitbart Juliana Freire Bharat Kumar Personalizing theWeb Using Site Descriptions.
30. M. Kifer, G. Lausen, and J. Wu. Logical foundations of object-oriented and frame-based languages. Journal of ACM, 42:741–843, July 1995.
31. The World Wide Web Consortium. Document Object Model (DOM) Level 2 HTML Specification, 2003. http://www.w3.org/DOM
32. Anupam, V. Freire, J. Kumar, B. and Lieuwen D. F. , Automating Web Navigation with the WebVCR, In Proceedings of WWW9, (Amsterdam, Netherlands,2000), Computer Networks, Vol.33, No. 1-6, p.503–517.
33. Mozilla , http://www.mozilla.org/.
34. JRex - The Java Browser Component,
http://jrex.mozdev.org/.
35. Liu L, Pu C, Han W. XWRAP: An XML-enabled Wrapper Construction System for Web Information Sources. In Proceedings of ICDE,pp, 611-621, 2000.
36. Flora-2 : http://flora.sourceforge.net.

Goal Programming and Scalarization Method for Solving Multiobjective Optimal Control Problem

M. Aidène and A. Merakeb

Department of Mathematics, Faculty of Sciences
University Mouloud Mammeri, Tizi-Ouzou, Algeria
aidene@mail.ummto.dz , merakeb_kader@yahoo.fr

Abstract. The multiobjective optimal control problem for a linear system with a bounded control action (input signal) and terminal constraints is considered. Some methods exist to solve a multiobjective problem, among these methods one knows the very used scalarization method and the Goal Programming theory. Using this two methods, we transform the multiobjective optimization problem into a minimization of a single objective by introducing unknown coefficients (weights) that will be used like variables in the optimization problem. We propose an iterative algorithm based on the block relaxation method for its resolution.

keywords : weakly efficient command, admissible control, multiobjective control problem, Goal Programming, scalarization method.

1 Introduction

Constrained optimization problems are of widespread interest because of their connection with optimal control and practical problems. In recent years, the analysis of several conflicting objectives has become an important issue. Such multiobjective optimization problems reflect the complexity of decision making in the real world. Of particular interest are performance indices expressible in linear form, for the obvious reasons that we consider the applications of LP software. Some of the most popularly used performance indices for linear control systems are the following:
(i) Minimum fuel problem

$$J_1 = \int_0^{t^*} u(t)dt.$$

(ii) Minimum amplitude problem

$$J_2 = \max_{t \in T} u(t).$$

(iii) Minimum final distance problem

$$J_3 = \|x(t^*) - x^*(t^*)\|_1$$

$$J_4 = \|x(t^*) - x^*(t^*)\|_\infty$$

where $x^*(t^*)$ is the given desired valued of $x(t^*)$.
(iv) Minimum integral absolute error problem

$$J_5 = \int_0^{t^*} \|x(t) - x^*(t)\|_1 dt$$

where $x^*(t)$ is the given demand vector.
Here the symbol $\|.\|$ denotes the norm of vector, i.e., for a vector $y = (y_1, ..., y_n)$, it is defined as $\|y\|_1 = \sum_{i=1}^{n} |y_i|, \quad \|y\|_\infty = \max_{1 \leq i \leq n} |y_i|$.

Multiobjective optimization problem optimizes a set of objectives with respect to the free controller parameters. Here the difficulty is that there is no unique optimal solution but rather a set of non dominated solutions. Many approaches are suggested in the literature to solve multiobjective optimization problems. They can be classified according three categories that depend on the construction of decision-maker's utility function. The first category of methods doesn't require any information on the decision-maker's preferences. The decision-maker only uses the axioms or concepts of optimality like the efficiency to characterize his/her optimal choice. Contrary to the others, this approach doesn't require any supposition and can be applied to all multiobjective optimization. The second approach is based on the idea that there exists an implicit function of utility exploited in the interactive methods. The third approach explicits directly the utility function that faithfully complies with the structure of preference of the decision-maker like scalarization methods and Goal Programming theory.

Goal Programming is a matter for a very advanced theory in the domain of the multiobjective problems. The general idea of the method is to establish a goal to reach for every criteria. Generally, the solution that satisfies all the goals (ideal point) is not feasible, the favorite solution would be therefore the one that comes closer the most possible of these goals. In the multiobjective problems with conflicting objectives, a Pareto solution can be obtained by solving a single objective problem with appropriate weighted utility function. Such approach leads to a solution of some other problem of which the offered solution can be considered like more or less acceptable for the original problem. Indeed, the choice of the scalar weights presents a problem in itself which is either relegated to decision-maker (how to choose these weights).

In this paper, it concerns the conversion of a linear multiobjective optimal control problem of linear dynamic system into a single objective problem obtained by scalarization and Goal Programming method and to consider the weighted coefficients as unknown parameters and use the block relaxation procedure for its resolution.

2 Problem statement

Consider the following multiobjective optimal control problem with terminal equality constraints in the class of piecewise continuous functions $u(.) = (u(t), t \in T = [0, t^*])$:

$$\max_u J(u) = \max_u Cx(t^*) = \max_u (c_k^t x(t^*), k = \overline{1,r}) \tag{1.1}$$

$$\dot{x} = Ax + bu, \qquad x(0) = x_0 \tag{1.2}$$

$$Hx(t^*) = g, \qquad d_* \leq u(t) \leq d^*, t \in T = [0, t^*] \tag{1.3}$$

where $C = (c_k^t, k = \overline{1,r})$ is the $r \times n$ matrix quality criteria ; $c_k = c_k(J)$, $k = \overline{1,r}$, is the $n-$vector costs of the k^{th} quality criterion ; $x(t) \in \boldsymbol{R}^n$ is the position of the system at time t ; $A = A(J,J)$ is an $n \times n$ matrix characterizing the dynamic of the system ; $b = b(J)$ is a given $n-$vector ; $H = H(I,J)$ is an $m \times n$ matrix of rank equal to $m \leq n$; g is a given $m-$vector representing the output signal at time t^* ; $u(t)$ is an operating command on the system bounded by d_*, $d^* \in \boldsymbol{R}$; $I = \{1, ..., m\}, J = \{1, ..., n\}$ are index sets ; symbol ^{t} denotes transposition.

Theorem 1. [9] *The system (1.2) is commandable if and only if the rank of the* $(n \times n)$ *matrix* $[b, Ab, A^2b, ..., A^{n-1}b]$ *is equal to* n.

By using the Cauchy formula, the solution of the system (1.2) can be written in the form

$$x(t) = F(t)[x_0 + \int_T F^{-1}(\tau) bu(\tau) d\tau] \tag{1.4}$$

where $F(t) = e^{At}, t \in [0, t^*]$ is the square $n \times n$ matrix defined by the relations $\dot{F}(t) = AF(t)$, $F(0) = I_n$ (I_n is the unit square matrix of order n).
By substituting this solution into the equations (1.1) and (1.3), we obtain the following equivalent formulation of the problem with the only variable $u(t)$:

$$\max_u J(u) = \max_u (CF(t^*)x_0 + \int_T C(t)u(t)dt) \tag{1.5}$$

$$\int_T \phi(t)u(t)dt = \overline{g} \tag{1.6}$$

$$d_* \leq u(t) \leq d^*, t \in T. \tag{1.7}$$

where, for $t \in T$

$\overline{g} = g - HF(t^*)x_0$
$p(t) = F(t^*)F^{-1}(t)b$
$C(t) = Cp(t)$
$\phi(t) = HF(t^*)F^{-1}(t)b.$

Definition 1. *A piecewise continuous function $u(t), t \in T$ is admissible if it satisfies the bound constraints (1.3) at each $t \in T$, and if there exists a trajectory $x(t), t \in T$ satisfying the dynamics (1.2) and the final state constraint (1.3).*

Remark 1. The problem consists in finding an admissible command u^0 corresponding with the optimal trajectory $x^0(t)$ that gives the maximum of the vector quality criteria $J(u)$. Contrary to the classical problems of optimization, where the usual order on $\boldsymbol{R}$ is total, in the multiobjective case, a solution that optimizes, simultaneously all criteria exists rarely. Considering the existing conflict between the criteria, the fundamental point in multiobjective problems is indeed the optimality concept. By conflict between two criteria J_1 and J_2, it is meant that for two commands $u, u^0 \in U$, we have $J_1(u) > J_1(u^0)$ but $J_2(u) < J_2(u^0)$.

Definition 2. *An admissible command $u^0 = u^0(t)$ is called weakly efficient (or Slater optimal) of problem* (1.1) *if there doesn't exist another admissible command $u = u(t)$ verifying $J_k(u) \geq J_k(u^0)$, for all k with strict inequality for at least one k.*

It means that from one weakly efficient situation, it is not possible to do a change to increase a criterion J_k without necessarily decreasing at least another criterion, while remaining on the whole of the admissible commands.

Definition 3. *Let $u^\varepsilon = u^\varepsilon(t)$ an admissible command. If for the weakly efficient admissible command $u^0 = u^0(t)$ there exists $k \in \{1, ..., r\}$ such that $J_k(u^0) - J_k(u^\varepsilon) \leq \varepsilon_k$ then u^ε is $\varepsilon-$optimal for criterion J_k.*
u^ε is called $\varepsilon-$efficient for the problem (1.1) *if*

$$J_k(u^0) - J_k(u^\varepsilon) \leq \varepsilon_k, \forall k = \overline{1, r}$$

(ε is an arbitrary real positive vector of dimension r).

It is not easy to determine the weakly efficient solutions while being based on their mathematics definitions. Even though the set of all efficient solutions is known, it is not clear which solution is better than the others. This is why it is useful to find an adequate definition of the optimality that is rigorous and comprehensible by the decision-makers and that the optimality in the case of single criterion would be a particular case. It is important that these methods are constructive, simple and feasible.

3 Scalarization and Goal Programming

The Goal Programming method is used to determine an efficient solution of a multiobjective problem. An efficient command is a non improvable solution.
The optimal control problems of linear dynamic system with terminal constraints and bounded control action are deeply studied in the literature [3], [5], the treatment of the multiobjective cases has known a large development. The problem

treated in [6] is however a multiobjective problem. It results from the methodology of Goal Programming that is used to determine an efficient solution (or weakly efficient) of one multiobjective problem. Its principle is to transform the vector functional $J(u)$ in a single minimizing function $K(u) = \|J^0 - J(u)\|$ in relation with a given norm $\|.\|$ of $\boldsymbol{R}^r$, with $J^0 = (J_k(u_k), k = \overline{1,r})$ and $J_k(u_k), k = \overline{1,r}$ are solutions of r problems

$$\max_u J_k(u), k = \overline{1,r}$$

under constraints (1.6) and (1.7).

$J_k(u_k)$ is a maximum (partial) according to an optimal command $u_k(t)$.
J^0 is called maximums partial vector (also called ideal point).

Let $\xi_k(u) = J_k(u_k) - J_k(u), k = \overline{1,r}$ for an arbitrary admissible command $u(t)$, then $\xi_k(u) \geq 0$. Indeed, $J_k(u) \leq J_k(u_k), \forall k = \overline{1,r}$.

$\xi_k(u)$ represents the regret to have chosen the command u to the prejudice of u_k for the criterion k. It is a value of deviation in relation to the partial maximum $J_k(u_k)$. It is obvious that the purpose is to minimize the regrets and to return the values of deviations $\xi_k(u), k = \overline{1,r}$ as small as possible.

One puts $\xi = J^0 - J(u)$ the vector of the regrets (deviations). If $\xi \neq 0$, that is to say there exists $\xi_k > 0$ (one can also write $\sum_{k=1}^r \xi_k > 0$), then two cases are to study:

1- the deviations $\xi(u)$ are improvable for which an admissible command $\overline{u}(t) = u(t) + \Delta u(t)$ verify $\xi_k(\overline{u}) \leq \xi_k(u), \forall k = \overline{1,r}$ with at least one strict inequality. What is equivalent to write $\xi(u + \Delta u) \leq \xi(u)$.

2- such deviations are non improvable in the sense where $\Delta u(t)$ doesn't exist. In this case the command u is efficient.

We have $K(u) = \|\xi\|$. For the Tchebychev norm $\|.\|_\infty$, one obtains the linear problem

$$\min_u K(u) = \min_u \max_{0 \leq k \leq r} \xi_k(u)$$

discussed in [6] under constraints (1.6) and (1.7).

Using the norm $\|.\|_1$, one obtains the also linear problem

$$\min_u K(u) = \min_u \sum_{k=1}^r (J_k(u_k) - J_k(u)) = \min_u \sum_{k=1}^r \xi_k(u).$$

Note that $K(u)$ consider each $\xi_k(u)$ as having the same importance in the expression of the regret. In the multiobjective problems, if the objectives have different importances levels, then a vector

$$\lambda \in \Lambda = \{\lambda \in \boldsymbol{R}^r, \lambda_k \geq 0, k = \overline{1,r}\}$$

can be chosen to represent the weights estimating the priorities between criteria. We define then the function L over two variables

$$L(\lambda, u) = \sum_{k=1}^{r} \lambda_k \xi_k(u) = \lambda^t.\xi(u)$$

what leads us to a scalarization problem that derives from a large advanced theory in the multiobjective problems of which the fundamental result is:

Theorem 2. [1] *If there exists $\lambda \in \Lambda$ for which $\lambda_k \neq 0$, for at least one $k = \overline{1,r}$ such that*

$$L(\lambda, u^0) = \min_u L(\lambda, u)$$

then u^0 is weakly efficient for the problem (1.1)-(1.3). This condition is sufficient for a linear problem.

Theorem 3. [1] *A weakly efficient solution u^0 is called without support if there doesn't exist $\lambda \in \Lambda$ for which u^0 optimizes the function $L(\lambda, u)$. In linear multi-objective problems, all weakly efficient solutions possess a support.*

Without loss of generality, one can assume that the vector λ is normalized relatively to the norm $\|.\|_1$ that is to say $\sum_{k=1}^{r} \lambda_k = 1$. The function $L(\lambda, u)$ is therefore a convex combination of the deviations $\xi_k(u)$.

The relative importance of the objectives is obviously a crucial information. Most methods translate this relative importance by numbers often named "weights" and the interpretation of these weights is not always immediate.
In the case where this information is impossible, one considers the coefficients λ_k as variables to optimize according to the expression

$$\min_{\lambda,u} L(\lambda, u) \tag{2.1}$$

under the constraints (1.6) and (1.7) and under the conditions

$$\sum_{k=1}^{r} \lambda_k = 1 \tag{2.2}$$

$$0 \leq \lambda_k \leq 1, \qquad k = \overline{1,r} \tag{2.3}$$

It is often helpful to partition the variables over which we are minimizing into two blocks of variables

$$\min_{\lambda} \min_{u} L(\lambda, u).$$

Block relaxation algorithms cycle through the blocks, minimizing over one block, while keeping the others fixed at their current values. At each iteration that is

$$\lambda^{(n+1)} = arg \min_{\lambda} L(\lambda, u^{(n)}),$$

$$u^{(n+1)} = arg \min_{u} L(\lambda^{(n+1)}, u).$$

Of course block relaxation in this case is interesting because the subproblems are easier to solve than the original problem. This can happen because of the structure of the constraint sets, but also because of the specification of the functional form.

4 The iterative algorithm

The algorithm used in [4] for the resolution of the problem (2.1) under constraints (1.6)-(1.7) using the adaptive method suggests an initial admissible command u^0. Let $(\lambda^{(n)}, u^{(n)})_{n\geq 1}$ be a sequence of iterates generated by the coordinate descent method, then $L(\lambda^{(n)}, u^{(n)})_{n\geq 1}$ is monotonically decreasing and bounded from below, then it converge. It comes when the set of the optimal solutions is nonempty [9].

Iterative algorithms are described by algorithmic maps

$$A : \Omega \to \Omega$$

which compute iterative sequences by

$$x^{(n+1)} = A(x^{(n)}).$$

The behavior of such sequences is described by one key theorem. This theorem address (global) convergence.
We remind the statement of Zangwill Convergence Theorem as reported in [12]. Given an optimization problem P on X, let M be the set of the points of X that fulfil a suitable necessary optimality condition. Suppose that, in order to solve P, it is used an algorithm represented by a point to set map $A : X \to X$.

Definition 4. *We say that $z : X \to R$ is a descent function (relative to the algorithm A) if it is continuous and has the following properties:*
1. $x \notin M$ implies $z(y) < z(x), \forall y \in A(x)$,
2. $x \in M$ implies $z(y) \leq z(x), \forall y \in A(x)$.

Theorem 4. [12] *Let P be an optimization problem on X and M be the set of the points of X that fulfil a certain necessary optimality condition. Let $A : X \to X$ be the algorithmic point to set mapping and consider a sequence $x^{(n)}$ generated by the algorithm, i.e. satisfying $x^{n+1} = A(x^{(n)})$. If the following three conditions hold:*
1. Every point $x^{(n)}$ is contained in a compact set $K \subset X$;
2. There exists a function of descent z;
3. The point to set map A is closed on X and $\forall x \in X, A(x) \neq \emptyset$.
Then, for every x which is the limit of a convergent subsequence of $x^{(n)}$, we have that $x \in M$.

5 The Adaptive Method

Consider the problem

$$\min_u L(\lambda, u) \tag{2.4}$$

under constraints (1.6)-(1.7), which the functional L is minimized with respect to a coordinate u (with the other coordinates λ held fixed). The applicability of the two-stage algorithm to solve this problem, discussed in [4], is usable while respecting the specification of the functional form. We describe in this section, the tools of this method while taking into account the construction of the functional L.

5.1 Support-control

We define a notion of support for the problem (2.4) under constraints (1.6)-(1.7) by choosing isolated times $\tau_i \in T, i = \overline{1,m}$ called moments of the support. The set $\tau_B = \tau_i, i = \overline{1,m}$ is called support if the corresponding matrix $\phi_B = (\int_{T_i} \phi(t)dt, i = \overline{1,m})$ is reversible.

$$T_B = \{T_i = [\underline{\tau_i}, \overline{\tau_i}] \subset T, [\underline{\tau_i}, \overline{\tau_i}] \cap [\underline{\tau_j}, \overline{\tau_j}] = \emptyset, i \neq j, i, j = \overline{1,m}\}$$

and $T_i = [\underline{\tau_i}, \overline{\tau_i}]$ with $\tau_i = \overline{\tau_i}$ or $\underline{\tau_i}$.
A pair $\{u, \tau_B\}$ formed from an admissible command u and a support τ_B is called support-control of the problem. The piecewise continuous function $u(t)$ will be discrete and takes its constant values on each interval T_i that is to say

$$u(t) = u_i, t \in T_i, \qquad d_* \leq u_i \leq d^*, i = \overline{1,m}$$

and

$$\phi_B = (\int_{T_i} \phi(t)dt, i = \overline{1,m})$$

is reversible.

As u is admissible then

$$\int_T \phi(t)u(t)dt = \overline{g}$$

and by using the support, we obtain

$$\int_{T_B} \phi(t)u(t)dt = \overline{g} - \int_{T_H} \phi(t)u(t)dt$$

with $T_B = \{T_i, i = \overline{1,m}\}$ and $T_H = T - T_B$.

Since $u(t)$ is constant by interval, we obtain

$$u(T_B) = (u_1, ..., u_m) = \phi^{-1}(T_B)(\overline{g} - \int_{T_H} \phi(t)u(t)dt).$$

5.2 Increment of the Functional

Let $\{u, \tau_B\}$ a starting support-control and $x(t)$ its corresponding trajectory. For $k = \overline{1,r}$, we construct the functions

$$\Delta_k(t) = c_k^t(p(t), t \in \tau_B)\phi_B^{-1}\phi(t) - c_k^t p(t), t \in T$$

Let $\overline{u}(t) = u(t) + \Delta u(t)$ another admissible command and $\overline{x}(t) = x(t) + \Delta x(t), t \in T$ its corresponding trajectory. The increment of the function

$$L(\lambda, u) = \sum_{k=1}^{r}(\lambda_k J_k(u_k) - \lambda_k c_k^t x(t^*))$$

is written by

$$\Delta L(\lambda, u) = L(\lambda, u) - J(\lambda, \overline{u}) = \sum_{k=1}^{r}(\lambda_k J_k(u_k) - \lambda_k c_k^t x(t^*)) - \sum_{k=1}^{r}(\lambda_k J_k(u_k) - \lambda_k c_k^t \overline{x}(t^*))$$

$$= \sum_{k=1}^{r} \lambda_k c_k^t \Delta x(t^*)$$

Using (1.4), we obtain

$$\Delta L(\lambda, u) = \sum_{k=1}^{r} \int_T \lambda_k c_k^t p(t) \Delta u(t) dt$$

Using the functions $\Delta_k(t)$, we obtain

$$\sum_{k=1}^{r} \int_T \lambda_k \Delta_k(t) \Delta u(t) dt = \sum_{k=1}^{r} \lambda_k c_k^t (p(t), t \in \tau_B)\phi_B^{-1} \int_T \phi(t) \Delta u(t) dt - \sum_{k=1}^{r} \lambda_k c_k^t \int_T p(t) \Delta u(t) dt.$$

From (1.6) we have $\int_T \phi(t) \Delta u(t) dt = 0$. From (1.4) we have $\Delta x(t^*) = \int_T p(t) \Delta u(t) dt$, then

$$\sum_{k=1}^{r} \lambda_k c_k^t \Delta x(t^*) = -\sum_{k=1}^{r} \int_T \lambda_k \Delta_k(t) \Delta u(t) dt$$

We put $\Delta(t) = \sum_{k=1}^{r} \lambda_k \Delta_k(t)$ and obtain

$$\Delta L(\lambda, u) = -\int_T \Delta(t) \Delta u(t) dt$$

5.3 Estimation of the Suboptimality Value

Let's form the following sets:
$T^+ = \{t \in T, \Delta(t) > 0\}$,
$T^- = \{t \in T, \Delta(t) < 0\}$,

From the admissibility of the new command $\overline{u}(t)$ we have:

$$d_* - u(t) \leq \Delta u(t) \leq d^* - u(t)$$

The maximum increment of the functional $\Delta L(\lambda, u)$ is reached for:

$$\begin{cases} \Delta u(t) = d_* - u(t), & \text{if} \quad t \in T^+ \\ \Delta u(t) = d^* - u(t), & \text{if} \quad t \in T^- \\ d_* - u(t) \leq \Delta u(t) \leq d^* - u(t), & \text{if} \quad \Delta(t) = 0 \end{cases}$$

and equal to

$$\beta(u, \tau_B) = \int_{T^+} \Delta(t)(u(t) - d_*)dt + \int_{T^-} \Delta(t)(u(t) - d^*)dt$$

called a suboptimality estimate value of the support-control $\{u, \tau_B\}$.

5.4 Optimality Criterion

The relations:

$$\begin{cases} u(t) = d_*, & \text{if} \quad \Delta(t) > 0 \\ u(t) = d^*, & \text{if} \quad \Delta(t) < 0 \\ d_* \leq u(t) \leq d^* & \text{if} \quad \Delta(t) = 0, t \in T \end{cases}$$

are sufficient for the optimality of the support-control $\{u, \tau_B\}$.

5.5 The $\varepsilon-$Optimality Principle

Let $u^o = u^o(t)$, $t \in T$, an optimal command of the problem (2.1) under constraints (1.6)-(1.7) and $u^\varepsilon = u^\varepsilon(t)$, $t \in T$, an admissible command where $\varepsilon \geq 0$ is a given value.

If $L(\lambda, u^o) - L(\lambda, u^\varepsilon) \leq \varepsilon$ then u^ε is called $\varepsilon-$optimal.

Remark 2. $\Delta L(\lambda, u) = L(\lambda, u^o) - L(\lambda, u^\varepsilon) \leq \beta(u^\varepsilon, \tau_B)$.

If $\varepsilon = 0$ then the $\varepsilon-$optimal principle becomes the optimal principle.

References

1. Steuer, R. (1986) Multiple Criteria Optimization: Theory, Computation and Application. Wiley, New York.
2. R. Gabassov, F.M. Kirrillova. Optimisation des systèmes lineaires. Edition Minsk, 1973 (en russe).
3. R. Gabassov, F.M. Kirrillova, O.I. Kostyukova. Adaptative method of solving large problem of linear programming. Preprints of IPAS-IFOPS Bulgaria 1979.
4. R. Gabassov, F.M. Kirrillova. Méthodes constructives d'optimisation. Problème de commande T2, edition de l'université de Minsk, 1984 (en russe).
5. O.I. Kostyukova. An algorithm for solving optimal control problems. Comput. Math. and Math. Phys.39, No 4, 545-559 (1999).

6. M.Aidene, I.L.Vorob'ev, B. Oukacha. Algorithm for solving a linear optimal control problem with minimax performance index. Comput. Math. and Math. Phys.45, No 10, 1691-1700 (2005).
7. M.M. El-Kady, M.S. Salim, A.M. El Sagheer. Numerical Treatement of Multiobjective Optimal Control Problems. Automatica, Volume 39, Issue 1, January 2003, Pages 47-55.
8. E.A. Galperin. Goal Optimal Pareto Solution of Multiobjective Linear Programs and its Computing with Standard Single Objective LP Software. Mathematical and Computer Modelling. 2003, Vol. 37, no 7-8, pp. 785-794.
9. V. Alexeev, V. Tikhomibov, S. Fomine. Commande Optimale. Edition Mir-Moscow, 1982.
10. G. Mastroeni. (1999) Minimax and Extremum Problems Associated to a Variational inequality. Rendiconti Del Circolo Matematico di Palermo, Serie II, Vol. 58, pp. 185-196.
11. P. Mahey. Decomposition Methods in Mathematical Programming. In Handbook of Applied Optimization, P. PARDALOS, M. RESENDE Eds, Oxford Press, 8 pages, (2002).
12. M. Minoux. (1986) Mathematical Programming, Theory and Algorithms, John Wiley, New York.

GAE M2LL and RFOS load balancing algorithms on dynamic networks : an experimental study

Jacques M. Bahi[1], Raphaël Couturier[1] and Abderrahmane Sider[2]

[1] Laboratoire d'Informatique de l'Université de Franche-Comté (LIFC). IUT de Belfort-Montbéliard, BP 527, 90016 Belfort Cedex, France.
{jacques.bahi,raphael.couturier}@iut-bm.univ-fcomte.fr
[2] Département d'Informatique, Université Abderrahmane Mira de Béjaïa, Route de Targa Ouzemmour Béjaïa 06000, Algérie. abd_sider@yahoo.fr

Abstract. Load balancing a distributed/parallel system consists in assigning work (load) to its processors so that they have to process approximately the same amount of it or amounts in relation with their computation power. The whole process is done according to some policy called a load balancing algorithm. A dynamic network is a distributed-memory parallel architecture with interconnecting network in which some processors and/or links can crash and recover or are slow. In this paper, we present a performance comparison between GAE (*Generalized Adaptive Exchange*) and RFOS (*Relaxed First Order Scheme*), two algorithms for dynamic networks in which only link failures are considered. The comparison is carried out on a dedicated test bed simulation framework that we have designed and implemented to this end. Simulation results show that although generating more communications, the GAE algorithm with the M2LL policy is faster than RFOS in balancing the system load. In addition, GAE M2LL is capable of achieving a more stable balanced state than RFOS.

Key words : load balancing, dynamic networks, performance comparison, Generalized Adaptive Exchange, Most to Least Loaded, Relaxed First Order Scheme.

1 Introduction

Solving large-size problems and reducing the execution times for small instances are the main purposes of parallel algorithms and architectures. Nowadays, the need for parallelism is becoming critical in many fields of science ranging from simulating fluid molecular dynamics and particle mechanics [9] to solving large optimization and scientific problems [7]. The data-parallel model for parallelization is based on splitting the data that has to be processed over several processing units. The amount of data that is allocated to processors has to be controlled because of two main reasons : the data associated with each processor may increase or decrease depending on the computation being carried out and the processors may have heterogeneous speeds. That is what makes load balancing

a fundamental problem that has to be addressed in the development of parallel/distributed software. It consists in allocating, according to a load balancing algorithm, some load to processors in relation with their computing powers. Load balancing algorithms have usually been characterized as static/dynamic, global/local, sender/receiver initiated and/or synchronous/asynchronous [20, 17, 21, 10].

Local load balancing algorithms are very attractive, because with this scheme, processors only know and use their direct neighboring load and this load is exchanged only with direct neighbors. These algorithms are iterative by nature, since they tend to balance the load *globally* (in the system) by successively balancing the load *locally* (in each neighborhood). Most popular local iterative algorithms are those named FOS (*First Order Scheme*) [11, 9, 23, 12, 13, 18] and their derived form called DE (*Dimension Exchange*) [11, 16, 22]. The difference between diffusion and dimension exchange lies in their ability to communicate with different nodes. If a node can perform simultaneous communication with its neighbors then diffusion is used to exchange load information and migration in parallel with all of them (in a single step of the algorithm). If not, then dimension exchange is to be done and in this case, a node exchanges load information and makes migration, cyclically, with each of its neighbors (only one per iteration). These algorithms were designed in a context of spreading usage of computing clusters formed by connecting machines by a rapid Ethernet-like network. But the recent evolution of network architectures toward the use of the Internet generates to a new execution environment for distributed computing. It is well known that the Internet is subject to contention and temporary link failures and processors may crash and recover. In this work, we deal with this concern supposing that the number of processors does not change and that a processor knows its living links and to what other processors it can send/receive messages. A link is alive if it can transmit a message in each direction [1]. FOS on dynamic networks has been investigated in [6, 14] and an accelerated version of it called RFOS appeared in [4]. Dimension Exchange on hypercube architectures with broken edges has been studied in [3]. Then studies researches have focused on adapting the most efficient DE-type algorithm named GDE (*Generalized Dimension Exchange*) [22] so that it could take into account broken edges. This enhanced version of GDE, called GAE (*Generalized Adaptative Exchange*), can be conducted according to several policies [2]. In [8], we presented a local, distributed and synchronous algorithm to implement the M2LL (*Most to Least Loaded*) policy. M2LL is used by the GAE algorithm for load balancing and aims at determining, for every iteration of GAE, all the pairs of nodes that will have to exchange work. This process has to ensure that the determined pairs are those that minimize the local imbalance of every neighborhood. In order to take into account broken edges at a given moment, we proposed the following solution : if a link is broken, then it is simply not considered for finding a pair. Systematically, each processor looks for a pair whatever edges are actually broken in the network, taking into account only neighbor nodes that had not chosen their pair yet.

In this paper, we plan to compare the GAE M2LL algorithm with the RFOS algorithm on dynamic networks. To the best of our knowledge, this work is the first to compare the two approaches of load balancing in dynamic networks. On static networks however, FOS and GDE load balancing algorithms have been broadly investigated (e.g. see [11,24]) and it is commonly agreed that GDE outperforms FOS in synchronous implementations.

The remainder of this paper is organized as follows. In Section 2, a detailed formulation of RFOS and GAE with M2LL policy is presented. This section is also devoted to define performance metrics used to compare our two algorithms. Section 3 first presents the simulation framework and then main and subsidiary results along with detailed discussions. Finally, Section 4 concludes this document and gives hints of our future work.

2 Preliminaries

A distributed-memory parallel system of n processors linked with an interconnection network is modelled by a graph $G = (V, E)$ where vertices V and edges of E represent the processors and links between them, respectively. Let E_B^t be the set of broken edges in the graph G at time (iteration) t and $N_i^t = \{j \in V : (i,j) \in E \wedge (i,j) \notin E_B^t\}$ the set of neighbor nodes of processor i at time t. The workload of node i is represented by a scalar non negative integer value w_i. At time t the system's load distribution is represented by the vector $w^t = \{w_1^t, w_2^t, w_3^t, \ldots, w_n^t\}$. The target of the load balancing process is to make this system load distribution converge towards the uniform load distribution represented by the vector $\overline{w} = \{\overline{w}, \overline{w}, \overline{w}, \ldots, \overline{w}\}$ where $\overline{w}$ is the load that every node should have received if a global knowledge of the overall system's load was known. If the system is constructed by assembling homogeneous processor powers and link bandwidths then $\overline{w} = \sum_{i=1}^{n} w_i/n$. In the other case, suppose each processors's power is represented by value s_i. Then node i should be allocated workload proportional to its power; that is $\overline{w_i} = \frac{\sum_{i=1}^{n} w_i}{\sum_{i=1}^{n} s_i} s_i$.

2.1 The Relaxed First Order Scheme

In [4], Bahi et *al.* presented an accelerated version of the well-known FOS (*First Order Scheme*) diffusion algorithm due to Cybenko in his fundamental contribution to iterative local load balancing algorithms [11]. According to this last scheme, on the hypercube static structure, a node i exchanges a portion noted α of its load difference $\mid w_j^t - w_i^t \mid$ with its neighbor j where $\alpha = \frac{1}{\Delta+1}$; Δ being the maximum degree of G. Later, FOS had been optimally tuned for some general static structures called k-ary n-cubes by Xu et al. [23]. The tuning of FOS is achieved by looking for a diffusion parameter noted α_{opt} is neither necessarily constant nor equal to $\frac{1}{\Delta+1}$. Moreover, authors gave general formulae to compute it for chain, ring, and mesh structures. Unfortunately, this is possible only for the so-called k-ary n-cubes static structures but not in dynamic network topologies, which often occur in real world distributed-memory architectures when some

communication links are broken even if the initial structure is supposed static. Boillat's independently devised FOS version [9] is another approach for choosing the diffusion parameter which becomes dependent in the degrees of two considered processors; i.e $\alpha_{i,j} = \frac{1}{max(d_i,d_j)+1}$ where d_i and d_j represent degrees of nodes i and j respectively. Anyway, by the FOS diffusion algorithm, the load evolution of processor i at time t is done according to formula 1 where $\alpha_{i,j}$ is fixed according to one way from the three above-cited ones.

$$w_i^t = w_i^{t-1} + \sum_{j \in N_i} \alpha_{i,j}(w_j^{t-1} - w_i^{t-1}) \tag{1}$$

The first adaptation of FOS for dynamic networks has been done in [6]. Authors showed that if the diffusion parameter is adapted to the current topology of the network by considering only living edges, then the FOS diffusion algorithm will converge under some realistic conditions : a necessary and sufficient condition is that no part of the network (one or several processors) should ever (i.e. during the process of iterative load balancing) be disconnected from the network. According to the new algorithm, the load evolution of processor i at time t is done according to formula 2 where $\alpha_{i,j}^t = \alpha_{ij}$ if $(i,j) \notin E_B^t$ and 0 if $(i,j) \in E_B^t$.

$$w_i^t = w_i^{t-1} + \sum_{j \in N_i^t} \alpha_{i,j}^t(w_j^{t-1} - w_i^{t-1}) \tag{2}$$

The RFOS diffusion algorithm (Relaxed First Order Scheme) [4] is an accelerated version of this latter algorithm. Acceleration is achieved again by introducing a relaxation factor noted β to the iterative formula 2 that results in a load evolving according to formula 3.

$$w_i^t = w_i^{t-1} + \beta \sum_{j \in N_i^t} \alpha_{i,j}^t(w_j^{t-1} - w_i^{t-1}) \tag{3}$$

The authors gave an exact formula for computing the optimal value of β noted β_{opt} that should be used to reach the maximum acceleration rate. Under the assumption that the total system load does not vary with time, they determined that at time step t, $\beta_{opt}^t = \min(R^t, \frac{2}{2-(s^t+l^t)})$ where s^t and l^t respectively represent the smallest and the largest diffusion matrix eigenvalue at time t and R^t is given by formula 4 that considers only processors i having $\sum_{j \in N_i^t} \alpha_{i,j}^t(w_j^{t-1} - w_i^{t-1}) < 0$.

$$R^t = min_i \frac{w_i^{t-1}}{(\sum_{j \in N_i^t} \alpha_{i,j}^t)(w_i^{t-1} - w_{min}^{t-1})} \tag{4}$$

The authors noted very opportunistically that even if -1 is always the smallest eigenvalue of the diffusion matrix (the connection graph is always disconnected), then RFOS will converge while FOS will not. From a practical point of view, we can add that RFOS with the *beta* parameter can be more suitable for dynamic

networks because a computing formula is known for it which is not the case for the optimal diffusion parameter of Xu et al. since they considered only static networks. In spite of this limitation, we still want to assess the use of an optimal diffusion parameter for dynamic networks if its value is initially computed when the computation is launched and then used throughout the overall load balancing steps without considering the impact of link failures.

2.2 The Generalized Adaptive Exchange with the M2LL Policy

Another approach for direct neighbor load balancing is to let a processor exchange load with only one of its neighbors at each load balancing step. Cybenko in [11] presented the DE (*Dimension Exchange*) algorithm on the structure of a hypercube interconnecting network. Under this scheme, a processor with id i will exchange load with some other node j (and vice versa) on a dimension of the hypercube noted d and computed according to the formula $d = (t\%D) + 1$ where t and D are respectively the current iteration number of the load balancing algorithm and the number of dimensions of the hypercube; hence the name of the algorithm. Formula 5 shows how the load of processor i evolves between iteration $t-1$ and t.

$$w_i^t = \frac{1}{2}(w_j^{t-1} - w_i^{t-1}) \tag{5}$$

Later, Hosseini et al [16] generalized DE for arbitrary interconnection networks by simulating the D dimensions with the maximum number of colors K necessary for coloring the edges of the graph representing the network topology. It is well known that their minimum number is bounded : $\Delta \leq K \leq (\Delta + 1)$ [15]. Load exchange takes place between two nodes i and j iff i and j are the extremities of some edge $e \in E$ having color $k = t\%K+1$. However, with DE and its generalized Hosseini version, the portion of load difference that is effectively exchanged is $\lambda = \frac{1}{2}$ and for this reason this scheme is called ADE (*Averaging Dimension Exchange*).

With GDE (*Generalized Dimension Exchange*)[22], Xu et al. proved that this would lead to the maximum convergence rate only on the hypercube. They go on and give an alternate way to fix the optimal value λ_{opt} for the k-ary n-cube structures like the chain, the ring, the mesh and the torus. Unfortunately again, the optimal exchange parameter is computed under the basic hypothesis of a static structure and consequently may not lead to the highest convergence rate on a dynamic network.

$$w_i^t = \lambda_{opt}(w_j^{t-1} - w_i^{t-1}) \tag{6}$$

The GAE (*Generalized Adaptive Exchange*) [2] is the adaptation of GDE for dynamic networks in which broken edges are considered. Authors showed that this can be done according to three different policies named arbitrary, random and most to least loaded (M2LL), all of which are only a particular case of FOS on dynamic networks and hence the convergence of GAE is guaranteed

under the same assumptions. The outcome of M2LL is a finite set of pairs such that each one represents two processors which have the greatest unbalance of their neighborhood. By making these two nodes exchange load, the objective is to tackle all such unbalances and thus to reach a balanced state as quickly as a pair can be formed. The GAE algorithm together with the M2LL policy can be seen as a per-iteration implementation of GDE. In fact, M2LL creates, when it is invoked (generally before applying GAE) a virtual ad hoc coloring in which, nodes of each dimension are chosen in such a way, that the load they will exchange will lead to a fast convergence to the uniform load distribution. Fixed pairs are different from one iteration to another according to load differences between nodes and available live links.

The reader can find detailed analysis and design of the M2LL Policy distributed algorithm in [8]. After having fixed a pair, a processor may apply formula 6 or consider an ADE-like exchange parameter in order to compute its load at time t. As for FOS, we want to assess the use of an optimal exchange parameter computed once on the initial static structure in the remaining iterations (despite broken edges) vs a default value that does not need computations.

2.3 Performance metrics

Three fundamental properties must be considered when assessing the performance of a load balancing algorithm : its termination, its efficiency and its stability [22]. Termination relates to the ability of the algorithm to lead any initial load distribution to the average load one. This is done mainly by means of formal proof. The efficiency is a subsidiary result of termination proof because it shows how many iterations or what execution time, the algorithm does spend until it reaches a load balanced state. Finally, stability describes the quality of the obtained global balance because the load is often not so evenly distributed and there still subsists an unbalance between undirectly linked nodes. In practice, this is modelled by some norm of the vector $\bar{w} - w^t$. The norm may be the Euclidean max norm noted $l_1 = max_i\{\mid \bar{w_i} - w_i \mid\}$ or the quadratic one $l_2 = (\sum_i (\bar{w_i} - w_i)^2)^{1/2}$ which will be used in this work. In addition, the amount of work loads moved over network's edges is an important criteria for a load balancing algorithm since it means cost of transmission and eventually less efficiency in the synchronous case.

3 Simulation and Results

This section is devoted to describe our simulation framework and then gives the main results concerning performance metrics defined in section 2.3.

3.1 Framework

Because we want to compare our two algorithms when using optimal diffusion and exchange parameters, we are somewhat conditioned in the initial static

networks to k-ary n-cube structures since these optimal parameters can be driven only for this class of topologies. So we shall focus on chain, ring, mesh, torus and hypercube interconnecting networks on architectures of 64 processors. These topologies differ by their maximum degree δ, their diameter D (see Table 11) and the number of links in the static situation and consequently their number in the dynamic case (see Table 1). Before every step of the simulation, a randomly generated dynamic topology is set up by breaking a subset of the static topology links; the subset size represents a percentage p of the edges of the considered network. We experiment three edge failure probabilities : 10, 30 and 50% of links are broken corresponding to the fact that the network respectively experiences low, mean and high link dynamics. Each result presented hereafter is the mean outcome of ten (10) distinct experimentations. The *opt* subscript associated with GAE M2LL or RFOS means respectively an optimal exchange parameter and an optimal diffusion parameter. Likewise the *cyb* expresses the use of a standard default value of the exchange parameter $\lambda = \frac{1}{2}$; while *boi* is for the use of Boillat's diffusion parameter. Table 2 shows standard and optimal exchange values for the initial static topologies and Table 3 shows Boillat and optimal diffusion parameter values for the same topologies along with the computed β relaxation factor value [19]- an empty entry mean that the value for this topology is variable. The global system's load is of 6400 virtual load units, meaning that the expected load average per processor is 100 per processor and initially the whole load is concentrated on processor with id 0. Virtual load units means that no load is actually transferred over edges, processes only update a local variable.

The test bed platform has been written in C++ and using the MPI message passing library. The Cartesian topologies feature of MPI has been very useful to emulate the chain, ring, 2D Mesh and Torus, 3D Mesh and the hypercube static architectures. All the experimentations had been carried out on an Intel Pentium IV CPU 512 MB desktop computer running Debian Gnu/Linux operating system under the LAM-MPI runtime environment simulating a 64 process architecture.

	Chain	Ring	2D Mesh	3D Mesh	2D Torus	Hypercube
Total Links p=0%	63	64	112	144	128	192
Broken Edges p=10%	7	7	12	15	13	20
Broken Edges p=30%	19	20	34	44	39	58
Broken Edges p=50%	32	32	56	72	64	96

Table 1. Number of total links for the 64-node initial static topologies and the number of broken edges for p= 10%, 30% and 50%

	Chain	Ring	2D Mesh	3D Mesh	2D Torus	Hypercube
λ_{cyb}	0.5	0.5	0.5	0.5	0.5	0.5
λ_{opt}	0.95	0.91	0.72	0.58	0.58	0.5

Table 2. Standard and optimal values of the exchange parameter for the 64-node experimented topologies

		Chain	Ring	2D Mesh	3D Mesh	2D Torus	Hypercube
Boilat	α	1/3	1/3	-	-	1/5	1/7
	β	1.5	1.496	1.263	1.197	1.164	1
Optimal	α	1/2	2/4.01	1/4	1/6	1/4.29	1/7
	β	1	1	1.019	1.108	1	1

Table 3. α standard and optimal values and corresponding β values for the 64-node static topologies

3.2 Convergence

Tables 4, 5 and 6 show the number of iterations before convergence is achieved in each neighborhood domain for a probability of edge failure of respectively 10, 30 and 50%.

	Chain	Ring	2D Mesh	3D Mesh	2D Torus	Hypercube
GAE_{cyb}	2915	935	133	49	40	19
GAE_{opt}	515	205	62	39	34	19
$RFOS_{boi}$	2536	790	117	43	39	18
$RFOS_{opt}$	2536	790	124	46	38	18

Table 4. Number of iterations for GAE and RFOS to converge at 10% of broken edges

	Chain	Ring	2D Mesh	3D Mesh	2D Torus	Hypercube
GAE_{cyb}	3675	1199	153	56	47	21
GAE_{opt}	1113	403	80	45	40	20
$RFOS_{boi}$	3223	1028	152	55	51	24
$RFOS_{opt}$	3220	1025	156	60	50	23

Table 5. Number of iterations for GAE and RFOS to converge at 30% of broken edges

	Chain	Ring	2D Mesh	3D Mesh	2D Torus	Hypercube
GAE_{cyb}	5079	1607	193	68	58	25
GAE_{opt}	1893	664	109	55	51	25
$RFOS_{boi}$	4637	1431	206	75	71	35
$RFOS_{opt}$	4625	1421	216	83	71	34

Table 6. Number of iterations for GAE and RFOS to converge at 50% of broken edges

We can notice that GAE M2LL with $\lambda = \frac{1}{2}$ is not faster than RFOS at 10% of broken edges but the more this rate increases, the more GAE M2LL competes with RFOS (on the 2D torus and the hypercube at 30% then additionally on the 2D and 3D Meshes at 50%). Let's focus now on convergence when using $\lambda = \lambda_{opt}$ computed once for the static topology and then used with no change through the load balancing process. Results show that GAE M2LL with an optimal exchange parameter is always faster than RFOS whatever the rate of broken edges is and the higher this latter is, the better GAE M2LL performs compared with RFOS and this remains valid for all kinds of topologies. On the other hand, RFOS with the Boillat or optimal diffusion parameter has approximately the same convergence rate. Notice also that with the Cybenco diffusion parameter, the number of iterations before RFOS converges, is far greater than with Boillat and Xu schemes and therefore results are not depicted at all.

3.3 Communication cost

The second metric we compare is the communication overhead. We focus on the migration phase for both algorithms where the communication overhead is measured by the total number of loads transferred on networks edges until convergence is established. Tables 7, 8 and 9 show total loads moved over networks' edges with a link failure probability of respectively 10, 30 and 50% .

	Chain	Ring	2D Mesh	3D Mesh	2D Torus	Hypercube
GAE_{cyb}	180272	97242	44239	28724	26061	21321
GAE_{opt}	323988	158306	47985	29439	28536	21597
$RFOS_{boi}$	180272	108123	43261	27890	27792	18782
$RFOS_{opt}$	180272	112728	73171	27819	27704	18769

Table 7. Load migration cost expressed as total load moved over networks' edges for RFOS and GAE M2LL with 10 % of broken edges.

	Chain	Ring	2D Mesh	3D Mesh	2D Torus	Hypercube
GAE_{cyb}	180272	97406	44440	29274	26892	21744
GAE_{opt}	418088	196213	52816	30332	29162	21801
$RFOS_{boi}$	180272	99130	43365	27923	25690	18801
$RFOS_{opt}$	180272	98291	43327	27849	25892	18817

Table 8. Load migration cost expressed as total load moved over networks' edges for RFOS and GAE M2LL with 30 % of broken edges.

	Chain	Ring	2D Mesh	3D Mesh	2D Torus	Hypercube
GAE_{cyb}	180272	97658	45139	29807	27800	22057
GAE_{opt}	461887	223133	56530	31550	30455	22178
$RFOS_{boi}$	180272	94120	43464	28009	25549	18866
$RFOS_{opt}$	180272	95929	43537	27884	25710	18854

Table 9. Load migration cost expressed as total load moved over networks' edges for RFOS and GAE with 50 % of broken edges.

Results show that GAE M2LL with optimal exchange parameter value implies always higher communication overhead. This is due to the exchange parameter value which is greater than $\frac{1}{2}$ which makes load travel many times over a network edge in both directions.

3.4 Stability

The stability metric exemplifies how evenly the load is distributed on processors by a load balancing algorithm. In order to measure it we use the Euclidian norm of two vectors : the vector of size 64 having in all its entries the mean expected load per processor i.e. 100 and the vector of system load actually achieved by the load balancing algorithm. Note that Table 10 just reports maximum values achieved over the three link failure rates because of data being very close.

	Chain	Ring	2D Mesh	3D Mesh	2D Torus	Hypercube
GAE_{cyb}	62	30	12	8	6	4
GAE_{opt}	60	29	11	8	6	4
$RFOS_{boi}$	62	62	38	39	28	30
$RFOS_{opt}$	62	62	38	40	28	30

Table 10. Maximum achieved stability with RFOS and GAE M2LL over all broken edge rates

These results show that the balance reached with GAE M2LL is better than with RFOS on all topologies. Moreover, with GAE M2LL, the maximum load difference of any two processors is bounded by the diameter D associated with the corresponding static topology (cf Table 11). On the other hand, with RFOS, the maximum load difference is equal to the diameter for the chain but is a multiple of it for the remaining topologies.

	Chain	Ring	2D Mesh	3D Mesh	2D Torus	Hypercube
Diameter D	63	32	14	9	8	6

Table 11. Diameter D of initial static topologies of 64 nodes

4 Conclusion

In this paper an experimental study and comparison between the GAE M2LL and RFOS load balancing algorithms on dynamic networks is presented. It comes out that the GAE M2LL load balancing algorithm is faster when using optimal exchange parameter values but leads to more communication overhead on networks edges. On the other hand, the balance reached after running GAE M2LL is bounded by the diameter of the initial network static topology. In perspective, we look forward to investigate asynchronous implementations of both algorithms and to integrate them into a real world scientific problem solving. Besides, a major direction for scientific computing is the use of Peer-to-Peer network consisting in a possibly infinite number of desktop computers, laptops and other computing devices (see [5]). Hence, a load balancing scheme supporting a varying number of nodes is appealing.

References

1. W. Aiello, B. Awerbuch, B. Zkfaggs, S.Rao, "Approximate load balancing on dynamic and asynchronous networks", *Proc. of the 25th annual ACM symposium on Theory of computing*, pages: 632–641, 1993.
2. J.M. Bahi, R. Couturier, and F. Vernier, "Load Balancing on Dynamic Network", *Technical Report RR-2002-1, LIFC, Université de Franche-Comté*, September 2002.
3. J.M. Bahi, R. Couturier, F. Vernier, "Broken Edges and Dimension Exchange Algorithm on Hypercube Topology", *Proc. of the 11th Euromicro Conference on Parallel, Distributed and Network-Based Processing (Euro-PDP'03)*, 2003.
4. J.M. Bahi, R. Couturier and F. Vernier, "Accelerated diffusion algorithms on general dynamic networks." *Proc. of 5th Int. Conference, PPAM Czestochowa, Poland*, LNCS Vol. 3019 : 77-82. PPAM, Springer-Verlag Heidelberg, 2003.
5. J.M. Bahi, R. Couturier, and P. Vuillemin, ŞJaceV : a programming and execution environment for asynchronous iterative computations on volatile nodes.Ť VECPARŠ06, Rio de Janeiro, Brazil, 2006.
6. J.M. Bahi and J. Gaber, "Load Balancing on Networks with Dynamically Changing Topology", *Europar Conference*, *LNCS* : 175–182, 2001.
7. J.M. Bahi, R. Couturier, P. Vuillemin ,"Solving nonlinear wave equations in the grid computing environment: an experimental study", in JCA (Journal of Computational Acoustics), 14(1), June 2006.
8. J.M. Bahi, R. Couturier and A. Sider, "Design and Analysis of the M2LL Policy Distributed Algorithm for Load Balancing in Dynamic Networks", *In Proc. of the 2006 Int. Symp. on Parallel and Distributed Processing and Applications (ISPA'06)*. LNCS Vol. 4331 : 195-204. Springer-Verlag Heidelberg, 2006.
9. J. Boillat, "Load balancing and poisson equation in a graph", *Concurrency: Practice and Experience*, 2(4):289-313, 1990.
10. A. Cortes, A. Ripoll, M.A. Senar and E. Luque, "Performance Comparison of Dynamic Load-balancing Strategies for Distributed Systems", *IEEE Proc. of the 32th Hawai Int. Conference on System Sciences*, Vol.8 : 8041–8051, 1999.
11. G. Cybenko. "Dynamic load balancing for distributed memory multiprocessors", *Journal of Parallel and Distributed Computing*, Vol. 7 :279-ŋ301, 1989.
12. R. Diekmann, A. Frommer, and B. Monien, "Efficient schemes for nearest neighbor load balancing" *Parallel Computing*, Vol. 25 (7) : 789ŋ-812, 1999.
13. R. Elsasser, B. Monien, and R. Preis, "Diffusion Schemes for Load Balancing on Heterogeneous Networks", *Theory of Computing Systems*, vol. 35, pp. 305–320, 2002.
14. R. Elsasser, B. Monien, S. Schamberger, "Load Balancing in Dynamic Networks", I-SPAN, 2004.
15. S. Fiorini and R.J. Wilson, "Edge-coloring of graphs", in *L.W. Beineke and R.J. Wilson, editors, Selected topics in graph theory*, Academic Press, 1978.
16. S. H. Hosseini, B. Litow, M. Malkawi, J. McPherson, and K. Vairavan, "Analysis of a graph coloring based distributed load balancing algorithm", *Journal of Parallel and Distributed Computing*, 10(2):160–166, Oct. 1990.
17. V. Kumar, G.Y. Ananth, V.N. Rao, "Scalable load balancing techniques for parallel computers", *Technical Report 91–55, Dept. of Computer Science, University of Minnesota*, 1991.
18. T. Rotaru and H.H. Nageli, "Dynamic load balancing by diffusion in heterogeneous systems", *Journal of Parallel and Distributed Computing*, 64 :481–497, 2004.
19. F. Vernier, "Algorithmique itérative pour l'équilibrage de charge dans les réseaux dynamiques", Phd thesis Université de Franche-Comté, 2004.

20. M. H. Willebeek-LeMair, A. p. Reeves, "Local vs. Global Strategies for Dynamic Load Balancing", *Proc. of the Int. Conference on Parallel Processing*, Vol. 1 : 569–570, 1990.
21. M. H. Willebeek-LeMair, A. p. Reeves, "Strategies for Dynamic Load Balancing on Highly Parallel Computers", *IEEE Trans. on Parallel and Distributed Systems*, Vol. 4 Nř 9 : 979-993, Septembre 1993.
22. C.Z. Xu and F.C.M. Lau, "Analysis of the Generalized Dimension Exchange Method for Dynamic Load Balancing", *Journal of Parallel and Distributed Computing*, 16:385–393, 1992.
23. C.-Z. Xu and F.C.M. Lau, "Optimal parameters for load balancing with the diffusion method in mesh networks", *Parallel Processing Letters*, 4(2) :139–147, 1994.
24. C. Xu, B. Monien, R. Lüling, F. C. M. Lau, "Nearest neighbor algorithms for load balancing in parallel computers", *Concurrency, Practice and Experience*, 7: 707–736, 1995.

Méthode de la diagonale dominante pour un problème d'optimisation globale en dimension n sur un hyperrectangle

Le Thi Hoai An[1] et Ouanes Mohand[1,2]

[1]Laboratoire de l'Informatique Théorique et Appliquée.
UFR MIM, Université Paul Verlaine-Metz, Ile du Saulcy. 57045 Metz, France.
Lethi@Univ-Metz.fr
[2]Département de Mathématiques.
Faculté des Sciences. Université Mouloud Mammeri, Tizi-Ouzou, Algérie
ouanes_mohand@yahoo.fr

Résumé Dans cet article nous présentons une méthode pour résoudre un problème d'optimistion globale sur un hyperrectangle de R_n. Les principales méthodes existentes utilisent (n+1) sommets pour construire la fonction borne inférieure (pour un minimum global), alors que notre méthode n'utilise que les deux sommets de la diagonale dominante .Des résultats de convergence sont montrés pour notre algorithme, qui utilise la méthode de Branch and Bound avec la w-subdivision exhaustive, qui est connue comme étant trés efficace par rapport aux autres subdivisions. Des exemples numériques sont traités pour montrer l'efficacité de notre méthode. Notre article est la généralisation de [12] .

Mots Clés : Optimisation globale ,fonction borne inférieure, branch and bound.

1 Introduction

On considere le problème suivant :

$$(P)\begin{cases}\min b(s)\\ s\in S\end{cases}$$

avec S un hyperrectangle de R^n et $b : S \to R$ une fonction une fois continument differentiable, même si les contraintes sont simples, ce problème reste trés difficile pour trouver la solution globale, il existe plusieures méthodes pour résoudre ce problème (voir [5], [6], [7], [8], [11] , [17]). En se basant sur [12], nous proposons dans ce travail une fonction borne inférieure convexe pour b, et un algorithme branch and bound pour résoudre (P) . La procédure de subdivision nous suggère d'utiliser la w-subdivision exhaustive, qui est montrée efficace dans les précédents travaux (voir [11] , [12] , [15]). Utilisant les deux sommets de la diagonale dominante de l'hyperrectangle, la nouvelle fonction borne inférieure est aussi quadratique, et peut être facilement calculée contrairement à celles proposées dans ([5] , [6] , [17]).

L'article est organisé comme suit : En 2 nous présentons la méthode branch and bound, tandis qu'en 3 nous présentons de nouveaux résultats concernant la fonction borne inférieure. L'algorithme et sa convergence sont étudiés en 4, et des exemples numériques sont traités en 5.

2 Méthode Branch and Bound

Elle consiste en une division successive d'un ensemble S en sous ensembles de plus en plus petits.A chaque sous ensemble de S,on construit une borne inférieure de la fonction objectif afin d'éliminer des parties ne contenant pas l'optimum global et de sélectionner le sous ensemble qu'on doit diviser.

Schéma général

Itération 0 :Initialisation
On pose $S = S_0$, on calcule s_0, LB_0, UB_0
si $UB_0 - LB_0 \leq \varepsilon$ stop s_0 est $\varepsilon-$optimale globale
sinon aller à l'iteration k

Itération k =1,2,3,.....
Etape 1 : Partition
Une partition de S_{k-1} est faite en utilisant la w-subdivision i.e le point s_{k-1} est utilisé pour diviser l'intervalle, si on est dans R ou bien si on a un hyperrectangle on divise suivant le plus grand coté en utilisant toujours s_{k-1}on obtient S_{k1}, S_{k2}

Etape 2 : Construction des bornes
on construit les bornes LB_{k1} , LB_{k2}, et on calcule UB_{k1} ,UB_{k2}

Etape 3 : Elimination
on pose $LB_k = \min(LB_{k1}$, LB_{k2} ,$LB_{k-1}^{res})$, $UB_k = \min($ UB_{k1} ,$UB_{k2}, UB_{k-1})$
Eliminer toutes les regions où $LB_{ji} \geq UB_k$; $j = 1, .., k$; $i = 1, 2$

Etape 4 : test d'arrêt
si $UB_k - LB_k \leq \varepsilon$ stop on a une solution ε-optimale globale
sinon continuer

Etape 5 : Selection
$\mathrm{LB}_k = \min(LB_{k1}$, LB_{k2} ,$LB_{k-1}^{res})$,soit s_k correspondant à ce minimum
$poserk = k + 1$,aller à l'Etape 1.

Remarque
La réalisation d'un algorithme de Branch and Bound dépend du choix des procédures suivantes :
−division des $\mathrm{S}_{k,i}$

- Sélection des $S_{k,i}$
- Estimation des bornes inférieures $LB_{k,i}$
- LB_{k-1}^{res} corespond aux bornes inférieures des intervalles qui ne sont ni subdivisés ni éliminés jusqu' à l'itération $(k-1)$.

3 Nouvelle borne inférieure

On propose une nouvelle fonction borne inférieure utilisant les deux sommets de la diagonale dominante c'est à dire le sommet inférieur le plus à gauche $\left(s^0\right) = (s_1^0, ..., s_n^0), s_i^0 \leq s_i; i = 1, ..., n$ et le sommet supérieur le plus à droite $\left(s^1\right) = (s_1^1, ..., s_n^1), s_i \leq s_i^1; i = 1, ..., n$ avec $s = (s_1, ..., s_n) \in S$

On considère la fonction borne inférieure suivante

$$\overline{LB}(s) = b(s^0)w_0(s) + b(s^1)w_1(s) + \Delta_d w_0(s).w_1(s)(l - L)$$

avec

$$l_i \leq \frac{\partial b(\xi)}{\partial s_i} \leq L_i; i = 1, .., n \ ; \forall \xi \in S$$

$$l = \min_{i=1,..,n} l_i \quad et \quad L = \max_{i=1,..,n} L_i$$

$$w_0(s) = \frac{\sum_{j=1}^n (s_j^1 - s_j)}{\sum_{j=1}^n (s_j^1 - s_j^0)} \quad et \quad w_1(s) = \frac{\sum_{j=1}^n (s_j - s_j^0)}{\sum_{j=1}^n (s_j^1 - s_j^0)}$$

$$\Delta_d = \sum_{j=1}^n (s_j^1 - s_j^0)$$

on a les propriétés des $w_i(s)$

$$0 \leq w_i(s) \leq 1; i = 0, 1$$

$$w_i(s^j) = \delta_{ij} = \begin{cases} 1, i = j \\ 0, i \neq j \end{cases}, i = 0, 1; j = 0, 1$$

$$\sum_{i=0}^{1} w_i(s) = 1$$

Remarque
Pour l'évaluation de l et L on utilise les techniques d'analyse intervalle.

Proposition 1

$$\overline{LB}(s) \leq b(s); \forall s \in S$$

Démonstration. on a d'abord $\overline{LB}(s)$ et $b(s)$ qui coincident aux sommets de la diagonale dominante s^0 et s^1, pour les autres points du domaine

on a

$$b(s) = b(s^0) + (s - s^0)\nabla b(\xi) \geq b(s^0) + \sum_{j=1}^{n}(s_j - s_j^0)l \qquad (1)$$

ou bien

$$b(s) = b(s^1) + (s - s^1)\nabla b(\eta) \geq b(s^1) + \sum_{j=1}^{n}(s_j - s_j^1)L \quad (2)$$

d'autre part

$$\begin{aligned}
\overline{LB}(s) &= \sum_{i=0}^{1} b(s^i)w_i(s) - \Delta_d\,(L - l)\,w_0(s)w_1(s) \\
&= (b(s^0) + l\Delta_d w_1(s))w_0(s) + (b(s^1) - L\Delta_d w_0(s))w_1(s) \\
&\leq \max\{b(s^0) + l\Delta_d w_1(s), b(s^1) - L\Delta_d w_0(s)\} \\
&\leq \max\{b(s^0) + \sum_{j=1}^{n}(s_j - s_j^0)l\ , b(s^1) + \sum_{j=1}^{n}(s_j - s_j^1)L\ \} \\
&\leq b(s)\ d'aprés\,(1)\,et\,(2)
\end{aligned}$$

donc $\overline{LB}(s) \leq b(s), \forall s \in S$.

Proposition 2
la fonction borne inférieure est quadratique

Démonstration. On a

$$\overline{LB}(s) = b(s^0)w_0(s) + b(s^1)w_1(s) + \Delta_d w_0(s).w_1(s)(l - L)$$

Calculons le produit

$$\begin{aligned}
w_0(s).w_1(s) &= \frac{1}{\Delta_d^2}\sum_{j=1}^{n}(s_j^1 - s_j).\sum_{j=1}^{n}(s_j - s_j^0) \\
&= \frac{1}{\Delta_d^2}(\sum_{j=1}^{n}s_j^1 - \sum_{j=1}^{n}s_j).(\sum_{j=1}^{n}s_j - \sum_{j=1}^{n}s_j^0) \\
&= \frac{1}{\Delta_d^2}[-\left(\sum_{j=1}^{n}s_j\right)^2 + \sum_{j=1}^{n}s_j[\sum_{j=1}^{n}s_j^0 + \sum_{j=1}^{n}s_j^1] - \sum_{j=1}^{n}s_j^1.\sum_{j=1}^{n}s_j^0]
\end{aligned}$$

la foction borne inférieure s'ecrit alors sous la forme

$$\begin{aligned}\overline{LB}(s) &= b(s^0)w_0(s) + b(s^1)w_1(s) + \Delta_d w_0(s).w_1(s)(l-L) \\ &= b(s^0)w_0(s) + b(s^1)w_1(s) + \\ &\quad + \frac{(l-L)}{\Delta_d}[-\left(\sum_{j=1}^{n} s_j\right)^2 + \sum_{j=1}^{n} s_j[\sum_{j=1}^{n} s_j^0 + \sum_{j=1}^{n} s_j^1] - \sum_{j=1}^{n} s_j^1.\sum_{j=1}^{n} s_j^0]\end{aligned}$$

donc elle est quadratique

Proposition 3
La fonction borne inférieure est convexe

Démonstration. La fonction borne inférieure est quadratique avec le coefficient de $\left(\sum_{j=1}^{n} S_j\right)^2$ qui est $\frac{(L-l)}{\Delta_d} > 0$ donc la matrice hessienne a tous ses éléments positifs égaux à $\frac{(L-l)}{\Delta_d}$toutes ses valeurs propres sont positives ou nulles donc elle est semi definie positive alors la fonction borne inférieure est convexe

4 Algorithme et convergence

4.1 Algorithme Branch and Bound

Nous allons mantenant décrire l'algorithme Branch and Bound pour résoudre le problème (P) .On note par LB_k, UB_k et s^k respectivement la meilleure borne inférieure ,la meilleure borne supérieure et la meilleure solution du problème (P) à l'itération k.

Algorithme

-Initialisation : Soit ε un nombre positif suffisemment petit ,calculer l et L

$$l = \min_{i=1,..,n} l_i, \;\; l_i \leq \frac{\partial b(s)}{\partial s_i}, \forall s \in S$$

et

$$L = \max_{i=1,..,n} L_i, \;\; \frac{\partial b(s)}{\partial s_i} \leq L_i, \forall s \in S$$

Poser $T^0 = S$ .Résoudre le problème convexe

$$\min\left\{\overline{LB}(s) \;\; : \;\; s \in T^0\right\} \qquad (3)$$

pour obtenir la solution $\overline{s}^0$

Poser

$$UB_0 = \min\{\min_{i=0,1} b(s^i), b(\overline{s}^0)\}; LB_0 = \overline{LB}(\overline{s}^0).$$

si $UB_0 - LB_0 \leq \varepsilon$ donc stop $\overline{s}^0$ est solution $\varepsilon-$optimale

sinon poser $M \leftarrow T^0, \quad k \leftarrow 1,$ et aller à l'itération k.

Itération k

$k1$. Poser

$$T^k = \prod_{i=1}^{n} \left[s_i^{1k}, s_i^{0k}\right] \in M$$

l'hyperrectangle tel que $LB_k = LB(T^k)$ et $\overline{s}^k$ la solution du problème (3).

$k2$.diviser T^k en deux sous-hyperrectangles T^{k_1}, T^{k_2} par la w-subdivision via $\overline{s}^k$.

$k3$. Pour $i = 1, 2$ faire

Résoudre le programme convexe (3) où k est remplacé par k_i pour obtenir $\overline{s}^{k_i}$

$k4$. actualiser la borne supérieure

$$UB_k = \min\{UB_{k-1}, b(\overline{s}^{k_1}), b(\overline{s}^{k_2})\}$$

Soit s^k la meilleure solution actuelle i.e $b(s^k) = UB_k$

Poser

$$M \leftarrow M \cup \{T^{k_i} : LB(T^{k_i}) < UB_k - \varepsilon, i = 1, 2\} \backslash \{T^k\}$$

Actualiser la borne inférieure :

$$LB_k = \min\{LB(T) : T \in M\}$$

$k5$. Si $M = \emptyset$ dons stop s^k est la solution optimale

sinon poser $k \leftarrow k + 1$ et retourner à $k1$.

4.2 Convergence

Théorème

L'algorithme Branch and Bound avec la w-subdivision exhaustive converge.

Démonstration. on a

$$\begin{aligned} 0 \leq b(s) - \overline{LB}(s) &= b(s^0) + (s - s^0)\nabla b(\xi) - \\ &\left(\sum_{i=0}^{1} b(s^i)w_i(s) - (L - l)\Delta_d w_0(s)w_1(s)\right) \\ &\leq b(s^0) + \sum_{j=1}^{n}(s_j - s_j^0)L - \min\{b(s^0) + \sum_{j=1}^{n}(s_j - s_j^0)l; b(s^1) + \sum_{j=1}^{n}(s_j - s_j^1)L\} \end{aligned}$$

Alors on a deux cas

1er cas

$$\min\{b(s^0) + \sum_{j=1}^{n}(s_j - s_j^0)l; b(s^1) + \sum_{j=1}^{n}(s_j - s_j^1)L\} = b(s^0) + \sum_{j=1}^{n}(s_j - s_j^0)l$$

alors
$0 \leq b(s) - \overline{LB}(s)$

$\leq b(s^0) + \sum_{j=1}^{n}(s_j - s_j^0)L - \left(b(s^0) + \sum_{j=1}^{n}(s_j - s_j^0)l\right)$

$\leq \sum_{j=1}^{n}(s_j - s_j^0)(L - l)$

$\leq \Delta_d(L - l) \to 0, \text{car}, \Delta_d \to 0.$dans Branch and Bound.

2eme cas
$\min\{b(s^0) + \sum_{j=1}^{n}(s_j - s_j^0)l; b(s^1) + \sum_{j=1}^{n}(s_j - s_j^1)L\}$

$= b(s^1) + \sum_{j=1}^{n}(s_j - s_j^1)L$
Alors

$0 \leq b(s) - \overline{LB}(s)$

$\leq b(s^0) + \sum_{j=1}^{n}(s_j - s_j^0)L - \left(b(s^1) + \sum_{j=1}^{n}(s_j - s_j^1)L\right)$

$= b(s^0) - b(s^1) + L(\sum_{j=1}^{n}(s_j - s_j^0) - \sum_{j=1}^{n}(s_j - s_j^1))$

$= (s^0 - s^1)\nabla b(\xi) + L(\sum_{j=1}^{n}(s_j^1 - s_j^0)$

$\leq (\sum_{j=1}^{n}(s_j^1 - s_j^0)(-l) + L(\sum_{j=1}^{n}(s_j^1 - s_j^0)))$

$= \sum_{j=1}^{n}(s_j^1 - s_j^0)(L - l)$

$\leq \Delta_d(L - l) \to 0, \text{car}, \Delta_d \to 0.$dans Branch and Bound.

Remarque
Dans Branch and Bound avec la w-subdivision exhaustive, on a une suite décroissante d'hyperrectangles qui tend vers un seul point donc la longueur de la diagonale dominante tend vers 0.

5 Exemples numériques

Exemple 1 [6]
$b(s_1, s_2) = -\sin(s_1)\sin(s_1 s_2)$ sur $[0, 4] \times [0, 4]$
la solution trouvée par notre méthode est $(1.5588, .99655)$ et $b(1.5588, .99655) = -.99978; LB = -1.1328$.
Avec pratiquement la même précision que dans [6]qui est de 0.1331
Le nombre d'évaluations de fonctions est égale à 99 qui est nettement inférieur

à celui de [6] qui est de 1013.

Exemple 2 [6]
$b(s_1, s_2) = 0.04(1 - s_1^2/4 - s_2^2) + (s_1 - 2s_2 + 1)^2/0.2$ sur $[1, 2] \times [1, 2]$
la solution trouvée par notre méthode est (1.7936,1.3772), b(1.7936,1.3772)=0.16905, LB=0.01401
Pour une précision de 0.155 qui est 2 fois meilleure que celle de [6] qui est 0.3, notre méthode n'a utilisé que 239 évaluations de fonctions contre 2673 pour [6].

Exemple 3 [6]
$b(s_1, s_2) = \left(s_2 - \frac{5s_1^2}{4\pi^2} + \frac{5s_1}{\pi} - 6\right)^2 + 10(1 - \frac{1}{8\pi}) \cos s_1 + 10$:sur $[-5, 10] \times [0, 15]$
la solution trouvée par notre méthode est (3.1389,2.2473) ;b(3.1389,2.2473)=.3979 ; $LB = .3847$
le nombre d'évaluations de fonction utilisé par notre algorithme est 447 pour une précision de 0.0132, contre 3519 pour une précision de 0.119 pour [6]

Sur ces trois exemples on voit que notre algorithme est plus efficace que celui proposé dans [6]

Conclusion. Nous avons proposé dans cet article une nouvelle fonction borne inférieure pour des fonctions une fois continument differentiables sur un hyperrectangle de R^net la méthode branch and bound pour minimiser ces fonctions avec des contraintes de "box". Notre fonction borne inférieure est déterminée avec seulement deux sommets de l'hyperrectangle, ceux de la diagonale dominante et elle est facilement calculée .La convergence de notre algorithme est démontrée et des exemples numériques sont traités pour montrer son efficacité.Il nous reste à valider numériquement notre algorithme sur les problèmes de grandes dimensions et la comparaison avec d'autres algorithmes existents, le travail dans cette direction est trés avancé.

Références

1. W. Baritompa and A.Cutler (1994), Acceleration for global optimization covering methods using seconde derivatives, Journal of Global Optimization, 4 : 329-341.
2. E. Baumann (1988), Optimal centered forms, BIT, 28 : 80-87.
3. C. de Boor (1978), A practical method Guide to Splines, Applied Mathematical Sciences, Spinger verlag.
4. P.G. Ciarlet (1979), The Finite Element Method for Elliptic Problems, Studies In Math.and its Appl.
5. V.P. Gergel (1997), A Global Optimization Algorithm for Multivariate functions with Lipschitzian First Derivatives, J.of Global Optimization, vol.10, N° 3, pp. 257-281.

6. E. Gourdin, B. Jaumard and R. Ellaia (1996), Global Optimization of Holder Functions, J.of Global Optimization, vol.8, N° 4, pp. 323-348.

7. E. Hansen (1980), Global optimization using Interval analysis the multidimensionnal case.Numer.Math., 34,pp. 247-270.

8. E. Hansen (1992), Global Optimization using Interval Analysis, Vol. 165 of Pure and Applied Mathematics, Marcel Dekker, New York.

9. R. Horst and P. M. Pardalos (1995), Handbook of Global Optimization, Kluwer Academic Publishers, Dordrecht.

10. R. Horst and H. Tuy (1993), Global Optimization, Deterministic Approaches, Springer Verlag, Berlin.

11. Le Thi Hoai An and Pham Dinh Tao (1998), A Branch and Bound Method via d.c. Optimization Algorithms and Ellipsoidal Technique for Box Constrained Nonconvex Quadratic Problems, J. of Global Optimization, vol 13, N° 2, pp. 171-206.

12. Le Thi Hoai An and Mohand Ouanes (2006), Convex quadratic underestimation and Branch and Bound for univariate global optimization with one nonconvex constraint, RAIRO Oper. Res. **40** : 285-302.

13. F. Messine and J. Lagouanelle (1998), Enclosure methods for multivariate diferentiable functions and application to global optimization, J.of Universal Computer Science, 4(6), pp. 589-603.

14. R. Moore (1966), Interval Analysis, Prence-Hall, Englewood Cliffs, New Jersey.

15. Thai Quynh Phong, Le Thi Hoai An and Pham Dinh Tao (1996), On the global solution of linearly constrained indefinite quadratic minimization problems by decomposition branch and bound method, RAIRO Oper. Res. **30 (1)**,pp. 31-49.

16. A. H. G. Rinnooy and G. H. Timmer (1989), Global Optimization, A Servey, In Nemhauser G.L. et al. eds., Handbooks of Operations Research1, North-Holland, Elsevier Science Publishers, pp. 631-662.

17. X. Wang and T. S. Chang (1998), A Multivariate Global Optimization Using Linear bounding Functions, J. of Global Optimization, vol 12, N° 4, pp. 383-404.

Etude de la transmission multicouche multicast avec différentiation de services

Hamdad Ismail, Elmagroud Rédouane,

Mekkakia. Maaza Zoulikha

Département d'Informatique, Université des Sciences et de la Technologie d'Oran

USTO, BP 1505 El M'Naoueur, Oran, ALGERIE

Tél/Fax +231 41425322

{i.hamdad@gmail.com , mekkakia@univ-usto.dz }

Résumé: Les travaux de recherche menés, dans cet article, consistent à analyser la réaction à la congestion dans une transmission multimédia multicouche assurée par le protocole RLM sur une topologie de réseau supportant le modèle de différentiation de service (DiffServ). Les différentes simulations montrent qu'avec DiffServ l'équité inter et intra protocole sont assurées. Et grâce à la séparation de l'aspect contrôle de l'aspect trafic, l'architecture DiffServ cohabite avec l'architecture multicast. Enfin cette étude montre que l'association des deux architectures DiffServ/RLM est possible. C'est une solution prometteuse pour assurer une qualité de service négociée.

Mots clés: DiffServ, Congestion, Multicast, RLM, QoS.

1. Introduction

Avec l'introduction de la qualité de service (QoS), l'Internet peut fournir d'autres classes de service que le traditionnel service best effort. De même la transmission multipoint est introduite pour partager des sessions entre les membres distants d'un même groupe. Par conséquent l'objectif de la QoS est de supporter des services de communications ayant des exigences qualitatives spécifiques.

Sachant que la Qos est obtenue par des protocoles aux systèmes terminaux, un des moyens les plus efficaces d'améliorer cette qualité de service est d'améliorer les protocoles aux systèmes terminaux; en particulier, les protocoles de contrôle de congestion.
Le protocole Received driven Layered Multicast (RLM) est le premier protocole de contrôle de congestion, non seulement pour la transmission vidéo sur internet, mais en général pour la transmission multimédias [8]. De plus, la différentiation de services (DiffServ) [2] est un modèle de services multiples qui peut répondre à des exigences de QoS.
L'objectif de cet article est l'étude de l'association de la transmission multicast multicouche, assurée par RLM, avec la différentiation de services.

2. Le modèle de différentiation de services (DiffServ)

La différentiation de service est une architecture IP QoS, basé sur le marquage des paquets, au moyen d'une information de priorité: Code d'accès aux services différentiés (DSCP). Pendant la période de la congestion, les paquets de priorité faible seront éliminés avant les paquets de haute priorité. Par conséquent, le modèle DiffServ (Differentiated services)[2] traite le trafic sous forme de classes de QoS (macro-flots). Les routeurs DiffServ ont des comportements de commutation de proche en proche (PHB) suivant la valeur de DSCP. Trois PHB sont définis : Best effort, Expedited Forwarding (EF), et Assured Forwarding (AF).
L'Acheminement Expédié (EF)[4] à été conçu, dès le départ, comme étant le composant d'un service bien spécifique : le service Premium [9], destiné aux applications demandant un délai et une gigue très faibles. EF est un comportement couteux.
La spécification de AF [6] définit 4 classes et 3 niveaux de priorité pour chaque classe. Chaque classe peut être vue comme une file d'attente séparée. Ceci permet le partage de ressources entre les flux, en réservant une classe AF pour chaque trafic. De ce fait, les flux seront isolés par l'algorithme d'ordonnancement dans les routeurs. Plusieurs méthodes pour attribuer une des trois priorités ont été proposées. La plus commune est le marqueur trois couleurs.
Le marqueur à trois couleurs, TCM, utilise deux seaux à jeton. Le premier sert à vérifier la conformité du flux avec le débit garanti (CR : Committed Rate) et le second à tester la conformité au débit crête (PR : Peak Rate). Si un paquet n'est pas conforme au second

seau à jeton il est marqué en rouge (la priorité de rejet la plus élevée). S'il est conforme au second mais pas au premier, le second niveau de priorité lui sera attribué : le jaune. Lorsque le paquet est conforme aux deux seaux à jeton il est marqué en vert (la plus faible priorité de rejet). Le TCM ainsi décrit ne tient pas compte de la couleur (priorité) du paquet à l'entrée. [7]. En manipulant les paramètres du TCM, il est possible de réguler le débit pour chacune des trois couleurs. Ainsi, en fonction des congestions dans le réseau, les paquets observeront des taux de pertes dépendant de leur couleur, suivant leur marquage. [7]

3. Le protocole RLM

RLM (Receiver-driven Layered Multicast) est le premier protocole de contrôle de congestion. C'est un protocole à couches cumulatives, non indépendantes, orienté récepteur [8]. La source créée un flux en couches, chaque couche apportant un raffinement par rapport aux précédentes. Chacune de ces couches sera associée à un groupe multicast. Le récepteur déterminera la qualité qu'il peut recevoir en s'abonnant graduellement à ces couches. Cette approche est extensible et résout le problème d'implosion de feed-back.

Le principe est encoder, découper et transmettre un flux multimédia en k couches cumulatives (C_1 ,C_2 , ..., C_{k-1} , C_k). La source émet, de façon indépendante, les diverses couches qu'elle génère et le récepteur sélectionne dynamiquement, parmi les couches générées, une collection de flux qui satisfasse les contraintes liées aux contrôles de flux et de congestion. A chaque couche on associe une adresse IP multicast particulière. Il suffit à un récepteur de souscrire aux adresses multicast qui correspondent au sous flux qu'il désire recevoir. En cas de congestion, le récepteur peut choisir d'abandonner une couche, en signifiant qu'il ne veut plus recevoir l'adresse multicast correspondante [8].

Pour mener à l'adaptation, un récepteur RLM doit déterminer si son niveau de suscription actuel est trop élevé ou bas. Par définition, le niveau est trop haut s'il cause une congestion, cette dernière est facilement détectable par la perte de paquets ou par la dégradation de la qualité.

D'autre part, quand le niveau de souscription est trop bas, le récepteur RLM doit ajouter spontanément les couches, à des moments qu'il considère propices, jusqu'à ce qu'il cause une congestion. Là il s'empresse de détruire la dernière couche ajoutée.

La souscription spontanée à la prochaine couche dans l'hiérarchie est appelée tentatives d'adhésion (join-experiment).
Or, ces tentatives peuvent causer une congestion transitoire qui peut détériorer la qualité du signal délivré. Pour éviter cela, RLM cherche à minimiser la fréquence et la durée des join-experiments sans que ça n'ait d'impact ni sur la convergence du protocole vers le bon débit, ni sur la capacité d'algorithme de réagir aux changements de l'état du réseau. Pour cela, chaque récepteur dispose sur chaque niveau de souscription d'un « join-timer » (temporisateur d'adhésion) qu'il multiplie par un facteur α (supérieur à 1) quand le niveau de souscription est un niveau problématique.
On définit le temps de détection (detection-time) par la durée nécessaire au réseau pour réagir à la tentative d'adhésion et pour que le récepteur en soit avisé. Si la tentative dure depuis plus longtemps que le temps de détection, sans qu'il n'y ait de congestion, alors le récepteur déduit que la tentative d'adhésion a réussi. Sinon la tentative aura échoué et le join-timer pour cette couche sera augmenté.

4. DiffServ et Multicast

DiffServ est une architecture orientée émetteur, tandis que multicast est orienté récepteur. Les routeurs du cœur d'un domaine DiffServ sont dégagés de tous mécanismes de contrôle, au contraire du multicast, où chaque routeur contrôle le groupe et maintient un état d'arbre multicast.
L'acheminement des trafics multicast, dans un domaine DiffServ, a rencontré des difficultés. Le problème NRS (Neglected Reserved Subtree) [3] résulte de la réplication des paquets multicast, ce qui engendre des nouveaux trafics dans le domaine DiffServ, d'où l'augmentation de la consommation des ressources par rapport a l'unicast. Puisque DiffServ fait un contrôle de trafic au niveau des routeurs de bordure et que ces trafics sont traités par agrégations, d'autres trafics peuvent alors souffrir d'élimination.
La solution proposée se présente sous forme d'une architecture orienté récepteur [1] qui consiste a séparer le plan contrôle du plan trafic, en plus d'utilisation de l'entité Bandwidth Broker (BB) pour faire le contrôle. Le BB a été proposé comme unité de gestion d'un domaine DiffServ [10][11], c'est un agent responsable de réserver les ressources demandées par les utilisateurs, et de configurer les routeurs. Il définit aussi des politiques basées sur la SLS (Service

Level Specification). Dans cette architecture, le BB traite les messages de contrôle multicast (join/leave) et il maintient un état de contrôle.

5. Simulation

5.1 Environnement de simulation

Nous avons utilisé le simulateur de réseau NS-2[5] de l'université de Berkeley/LBL (Lawrence Berkeley Laboratory) pour simuler le comportement des sources protocolaires RLM et leur équités inter et intra protocoles dans un réseau hétérogène avec et sans différentiation de services.
A partir d'une configuration (nœuds et liens), nous avons programmé des scripts en TCL sous NS en définissant toutes les propriétés du réseau : débit, flux, files d'attente, temps de départ … et la durée de simulation.
Les résultats des simulations sont dans des fichiers contenants les événements constatés sur le réseau. Nous avons utilisé ces fichiers pour construire les graphes.

5.2 Description de la simulation

Nous simulons le comportement des sources protocolaires, dans le cas d'un réseau hétérogène avec et sans DiffServ. Le réseau (figure 1) est composé de huit nœuds, avec deux sources qui sont en concurrence sur un lien central, qui fait office de goulot d'étranglement.
Les nœuds 0,1 sont des sources et le nœud 6 et 7 sont les destinations.
Les capacités des nœuds ont été choisies de telle façon que les pertes se produisent uniquement sur le lien central. Les liens sont de 10 Mbit/s et le goulot d'engorgement est de 5 Mbit/s.
Les paquets TCP émis sont de 1 Ko. Une source RLM émet sept couches à un débit de 128, 256, 512, 1024, 2048, 4096, 8192 Kbit/s respectivement pour chaque couche.
Les constantes de RLM ont les valeurs suivantes : α=2, β=2/3, g_l=0.25, $\overline{T_J^{min}}$=5sec, $\overline{T_J^{max}}$=60sec.
Pour la différentiation de service les PHB utilisés sont les AF, Le marquage est effectué indépendamment pour chaque source et destination en utilisant le TCM.

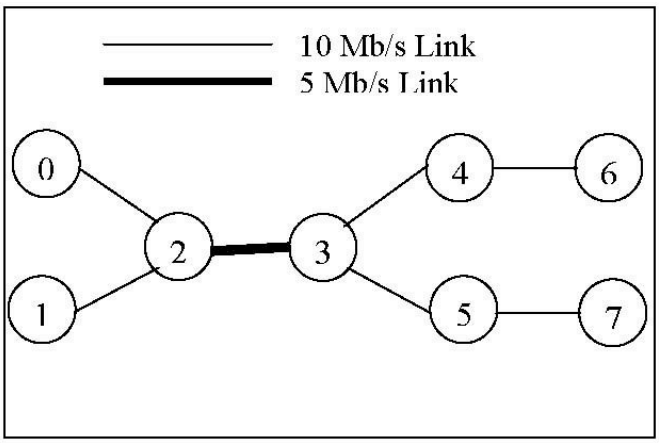


Figure 1: topologie de simulation

Le but de cette phase de simulation est l'étude de la réaction à la congestion des récepteurs RLM et de l'équité inter et intra protocoles dans une topologie de réseau hétérogène avec et sans la différentiation de services.

5.3 Equité intra protocole

Les sources 0,1 émettent des flux CBR sur UDP avec RLM.
Le nœud 6 (Récepteur RLM) rejoint les sessions RLM du nœud 0.
Le nœud 7 (Récepteur RLM) rejoint les sessions RLM du nœud 1.

5.3.1 Comportement des récepteurs RLM dans un réseau sans DiffServ

Les nœuds 2, 3, 4 et 5 ont un comportement de routeur classique. La figure 2 montre les résultats des simulations pour les deux récepteurs RLM (nœud 6 et nœud 7).

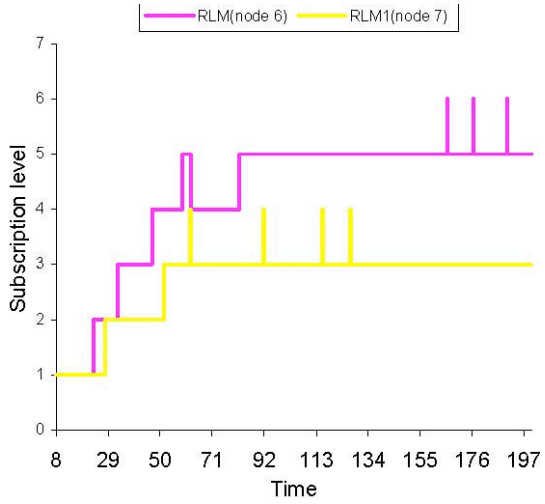


Figure 2 : Niveaux de souscriptions des récepteurs RLM sans DiffServ.

De la figure 2 nous remarquons que l'un des récepteurs RLM (nœud 6) a essayé de se souscrire avec échec à la couche 5 à environ 59s, il a réussit à se stabiliser à cette couche dans sa deuxième tentative à environ 82s ;Il fait trois tentatives pour se souscrire à la 6e couche entre 166 et 190s sans succès.

Le deuxième récepteur RLM (nœud 7) ne s'est stabilisé qu'à la 3ème couche dés 52s, avec 4 tentatives de souscription à la 4ème couche entre 62 et 127s sans succès.

De plus nous remarquons que le premier abandon de la 5ème couche par le récepteur (nœud 6) est due à la tentative de souscription de l'autre récepteur à la 4ème couche, ce qui a causé la congestion du goulot d'engorgement, donc l'abandon des couches en cours d'expérimentation. Nous remarquons que le récepteur (nœud 6) est mieux servi, alors qu'il est identique au récepteur (nœud 7).

5.3.2 Comportement des récepteurs RLM dans un réseau avec DiffServ

Les nœuds 2, 4 et 5 sont des nœuds de bordure du domaine DiffServ (DS); le nœud 3 représente un nœud de cœur du DS. Les paquets RLM sont marqués de la même classe AF avec même précédence initiale pour le TCM. La figure 3 montre les résultats des simulations pour les deux récepteurs RLM (nœud 6 et nœud 7).

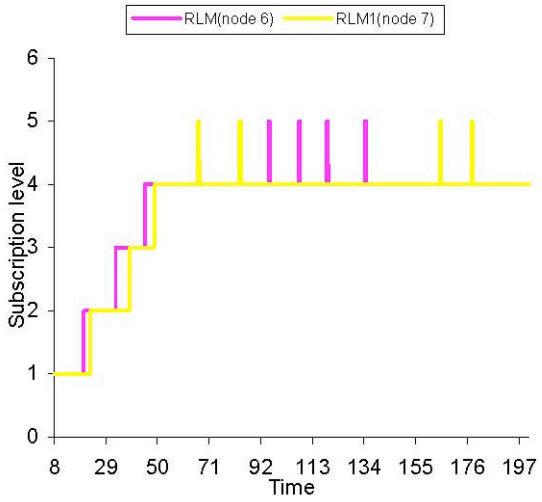


Figure 3 : Niveaux de souscriptions des récepteurs RLM avec DiffServ.

De la figure 3; nous remarquons que le récepteurs (nœud 6) se souscrit à la 4ème couche à l'instant 42s avec 5 tentatives de souscription sans succès à la 5ème couche.
Le deuxième récepteur RLM (nœud 7) se stabilise aussi à la 4ème couche dés 46s, avec 4 tentatives de souscription à la 5ème couche.
Les deux récepteurs RLM se stabilisent à la même couche avec un léger différé pour les souscriptions, ils sont presque équitablement servis.

5.3.3 Synthèse

Des ces deux simulations, nous concluons que, dans un réseau sans DiffServ, l'équité intra protocole pour RLM n'est pas assurée. DiffServ assure une certaine équité entre les sessions RLM, du fait qu'il garanti un même traitement pour des flux de même type.

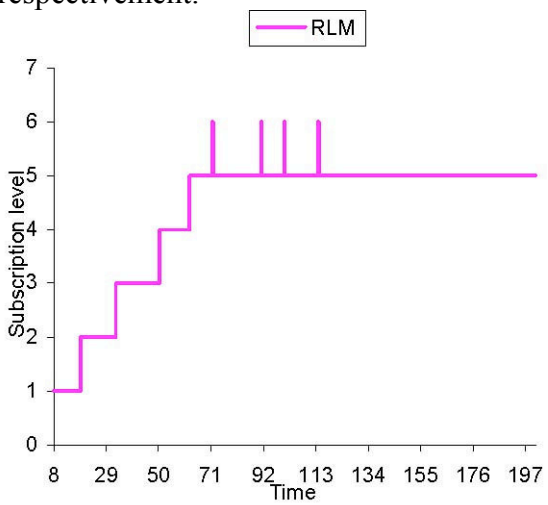
5.4 Equité inter protocole

La source 0 émet des flux CBR sur UDP avec RLM.
La source 1 émet un flux FTP sur TCP vers le nœud 7.
Le nœud 6 (Récepteur RLM) rejoint les sessions RLM du nœud 0.

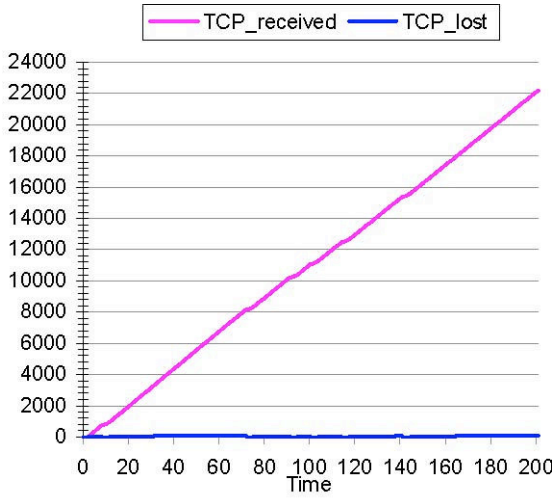
5.4.1 Comportement RLM/TCP dans un réseau sans DiffServ

Les nœuds 2, 3, 4, 5 ont un comportement de routeurs classiques.
Les figures 4.a et 4.b montrent les résultats de simulation pour le

récepteur RLM (nœud 6) et le récepteur TCP (nœud 7) respectivement.



a. Souscription du RLM



b. Comportement du TCP

Figure 4 : RLM/TCP sans DiffServ.

De la figure 4.a, nous remarquons que le récepteur RLM s'est souscrit à la 5ème couche à l'instant 62s avec quatre tentatives sans succès de souscription à la couche 6 entre l'instant 72 et 114s ; Il se stabilise à la 5ème couche.

De la figure 4.b, nous remarquons que le taux du flux TCP reçu atteint les 22 210 paquets avec une retransmission des paquets perdus (négligeables) qui sont dus aux tentatives de souscription du récepteur RLM à la 6ème couche.

5.4.2 Comportement RLM/TCP dans un réseau avec DiffServ

Les nœuds 2, 4 et 5 sont des nœuds de bordure du domaine DiffServ DS ; le nœud 3 représente le nœud de cœur du DS. Quatre scénarios d'attribution des PHBs ont été simulés:
- Les flux RLM et TCP affectés à deux classes AF différentes:
Le récepteur RLM se souscrit à la 5 ème couche dés l'instant 71s et se stabilise à 96s.
Le flux TCP reçu atteint les 23 751 paquets avec une retransmission des paquets perdus.
-Les flux RLM et TCP affectés à la même classe AF et marqués de la même précédence initiale :
Le récepteur RLM se souscrit à la 5ème couche à l'instant 56.
Le flux TCP reçu atteint les 21 890 paquets avec une retransmission des paquets perdus (négligeables).
- Les flux RLM et TCP affectés à la même classe AF et le flux TCP est marqué initialement dans la troisième précédence qui correspond à la file red du TCM, donc les flux RLM appartenant à la première et deuxième précédence (les plus prioritaires) sont privilégiés :
Le récepteur RLM se souscrit à la 5ème couche à l'instant 48s. Il se stabilise à la 5ème couche.
Le flux TCP reçu atteint les 21275 paquets avec une retransmission des paquets perdus (négligeables).
- Les flux RLM et TCP sont affectés à la même classe AF et les flux RLM sont marqués initialement dans la troisième précédence qui correspond à la file red du TCM, donc les flux TCP appartenant à la première et deuxième précédence (les plus prioritaires) sont privilégiés :
Le récepteur RLM se souscrit à la 5ème couche à l'instant 60.
Le flux TCP reçu atteint les 22375 paquets avec une retransmission des paquets perdus (négligeables), toujours due aux tentatives de souscription du récepteur RLM à la 6ème couche.

5.4.3 Synthèse

Par rapport à la topologie sans DiffServ on constate les résultats suivants :
- Avec deux classes AF il y a gain : de 1541 paquets TCP, et une retard de 38 secondes pour la souscription du récepteur RLM à la couche de stabilisation.

- Avec une classes AF et même priorité RLM/TCP il y a une perte de 320 paquets TCP, et un gain de 6 secondes pour la souscription du récepteur RLM à la couche de stabilisation.
- Avec une classe AF, RLM prioritaire, il y a une perte de 935 paquets TCP et un gain de 14 secondes pour la souscription du récepteur RLM à la couche de stabilisation.
- Avec une classe AF, RLM défavorisé, il y a un gain de 166 paquets TCP et un gain de 2 secondes pour la souscription du récepteur RLM à la couche de stabilisation.
- La solution avec deux classes AF est intéressante, mais c'est une solution coûteuse, en terme de ressources.

Les autres solutions ne nous permettent pas d'allier l'optimisation RLM et TCP.

6. Conclusion

Les couches RLM sont considérées comme des flux indépendants possédant chacune un débit spécifique. Vu que nous les avons marqués de la même classe AF et même précédence initiale pour le TCM, DiffServ assure une même qualité de services pour ces flux. Donc les couches de niveaux supérieurs (à haut débit) sont susceptibles de ne pas satisfaire les critères de la précédence Green du TCM, ce qui les rend plus exposés aux pertes. Au contraire, dans le comportement sans DiffServ les pertes sont aléatoires pour toutes les couches. Les résultats des simulations effectuées affirment que la différentiation de service (DiffServ) assure une certaine équité entre les sessions RLM, d'où l'équité intra-protocoles.

Pour l'équité inter-protocoles, l'affectation d'une classe à chaque type de flux RLM et TCP représente une solution car le comportement pathologique lié à RLM dans l'Internet est dû au conflit entre les mécanismes de contrôle de congestion de RLM et du TCP. Le problème est que les congestions TCP sont considérées comme cogestion RLM par les récepteurs RLM, ce qui cause les abondons des souscriptions aux couches en cours d'expérimentation, même si elles ne sont pas la cause de cette congestion. Donc, le fait d'utiliser une classe pour chaque type de flux permet de résoudre ce conflit, car chaque classe est traitée indépendamment.

Toutefois, l'utilisation d'une seule classe AF pour RLM et TCP ne nous permet pas d'allier l'optimisation des deux flux, il faut donc choisir le modèle de priorité selon la qualité de service négociée.

Enfin, grâce à la séparation de l'aspect contrôle de l'aspect trafic, la cohabitation de l'architecture DiffServ avec l'architecture multicast est possible. C'est une solution prometteuse pour assurer une QoS négociée.

Références

[1] F. Aiguo, M. Gerla, Receiver Initiated Multicasting with Multiple QoS Constraint, In Proc. Infocom 2000.

[2] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss An architecture for Differentiated Services, RFC 2475, IETF, December 1998

[3] R. Bless, K. Wehrle, IP Multicast in Differentiated Services Networks, IETF Internet Draft, draft-bless-DiffServ-multicast-06.txt, Février 2003

[4] B. Davie, A. Charny, J.C.R. Bennett, K. Benson, J.Y. Le Boudec, W. Courtney, S. Davari, V. Firoiu, D. Stiliadis An expedited forwarding PHB (Per-Hop Behavior), RFC 3246, IETF, Mars 2002

[5] Kevin Fall, Kannan Varadhan, The ns Manual, avril 2003.

[6] J. Heinanen, F. Baker, W. Weiss, J. Wroclawski, Assured forwarding PHB group, RFC 2597, IETF, Jun 1999

[7] J. Heinanen, R. Guerin, A Two Rate Three Color Marker, RFC 2698, Jun 1994.

[8] Steven McCanne, Van Jacobson Martin Vetterli, RLM (Receiver-driven Layered Multicast) ACM SIGCOMM'96 August 1996, Stanford, CA.

[9] K. Nichols, V. Jacobson, L. Zhang A Two-bit Differentiated Services Architecture for the Internet, RFC 2638, IETF, Juil 1999

[10] Z.-L. Zhang, Z. Duan, L. Gao and Y.T. Hou Decoupling QoS Control from Core Routers: A Novel Bandwidth Broker Architecture for Scalable Support of guaranteed Services, Proc. ACM SIGCOMM, pp.71-83, Stockholm, Sweden, Aug. 2000.

[11] Z.-L. Zhang, Z. Duan, and Y.T. Hou "On Scalable Design of Bandwidth Brokers", IEICE Transaction on Communications,E84-B (8), Aug. 2001.

An Other Characterization of An Induced Subgraph of An Hypercube

Abdelhafid BERRACHEDI and Nawel KAHOUL

Faculté de Mathématiques, USTHB,
BP 32 El-Alia, Bab-Ezzouar 16111, Alger, Algérie
aberrachedi@usthb.dz,
kahoul.nawel@caramail.com

Abstract. A conjecture of *I.Havel* asserts that there is a Hamilton cycle for the graph induced by the middle levels of any hypercube of an odd degree. In this paper, we are interested in giving some new characterizations for this graph in the class of graphs for which each path of length three belongs to exactly one single cycle of length six.

Key Words : Hypercubes, Hamming graphs, Odd graphs, semi-regular graphs, cycle regular graphs, (0,λ)-graphs

1 Introduction

Unless specified otherwise, all graphs in this paper are finite, simple, undirected and connected. A graph G will have a vertex set $V(G)$ and an edge set $E(G)$; sometimes we will write simply $G = (V, E)$. The graph on n pairwise adjacent vertices is denoted by K_n. The *neighbourhood* of a vertex $u \in V(G)$ will be denoted by $N(u)$. The *degree* of a vertex u of G and the *minimum degree* of vertices of G will be respectively denoted by $d(u)$ and $\delta(G)$. A bipartite graph is *semi-regular*, if the vertices in the same bipartition have the same degree. An *elementary path* $P_{\mu+1}$ of length μ (in G), is a sequence $u_0 \cdots u_\mu$ of pairwise distinct vertices except possibly u_0 and u_μ, where $u_iu_{i+1} \in E(G)$, for $i = 0, \cdots, \mu - 1$; it will sometimes be called a (u_0, u_μ) -path. An *elementary cycle* of length μ (in G) is a (u_0, u_μ)-path with $u_0 = u_\mu$ and is denoted *μ-cycle*. Both a (u_0, u_μ)-path and a μ-cycle are *induced* if $u_iu_j \notin E(G)$, whenever $|i - j| \neq 1$(substraction modulo $\mu + 1$). The *girth* of graph G is the length of the shortest cycle in G. A (u_0, u_μ)-path *belongs* to an elementary cycle $v_0 \cdots v_{\nu-1}v_0$, if $\mu \leq \nu$ and $u_0 = v_i$, for some $0 \leq i \leq \nu - 1$.

The *distance* between two vertices u and v in G is the length of the shortest (u, v)-path and is denoted by $d(u, v)$. The *diameter* of the graph G is $diam(G) = max\{d(u, v) : u, v \in V(G)\}$. For a vertex $u \in V(G)$, define $N_i(u) = \{v \in V(G) : d(u, v) = i\}$. For a given $u \in V(G)$ and a positif integer n, such that $n = max_{v \in V(G)} d(u, v)$, the non-empty sets $N_i(u)(i = 0, \cdots, n)$ define a partition of $V(G)$ which we shall call *level decomposition from* u and the set $N_i(u)$ will be called the i^{th} *level* (in this paper, we shall write N_i instead of $N_i(u)$). Given

$u \in V(G)$ and the level decomposition from u, define, for $v \in N_i$, $d^-(v) = |N(v) \cap N_{i-1}|$ and similarly, $d^+(v) = |N(v) \cap N_{i+1}|$. We will sometimes refer to $d^-(v)$ and $d^+(v)$ as $m(v, N_{i-1})$ and $m(v, N_{i+1})$, respectively. The *Categorical product* $G \times H$ of two graphs G and H has as vertex-set $V(G \times H) = V(G) \times V(H)$ and where two vertices (u, v) and (u', v') in $G \times H$ are adjacent if and only if $(uu' \in E(G)$ and $vv' \in E(H))$.

The *hypercube* Q_n has $V = \{A : A \subseteq \{1, 2, \cdots, n\}\}$ as vertex-set and two vertices A and B are adjacent if and only if $|A \Delta B| = |(A \setminus B) \cup (B \setminus A)| = 1$. The hypercube is regular of degree n and has diameter n.

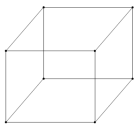


Fig. 1. Cube Q_3

The subgraph of Q_n induced by the two consecutive levels N_{k-1} and N_k and denoted by L_n^k is of degrees $n - k + 1$ and k, of order $\binom{n}{k} + \binom{n}{k-1}$. In particular, the subgraph induced by the two middle levels N_{k-1} and N_k of the hypercube Q_{2k-1} and denoted by L_{2k-1}^k or more frequently H_k, is regular of degree k. For $k = 3$, we obtain the Desargues graph H_3.

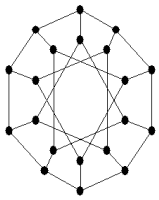


Fig. 2. Desargues Graph H_3

The Odd graph O_n has the set $\{A : A \subseteq \{1, 2, \cdots, 2n-1\}; |A| = n-1\}$ as vertex-set and two vertices are adjacent if their corresponding subsets are disjoint. The odd graph O_n is regular of degree n and for $n \geq 4$; O_n is of girth six.The *Extended Odd graph* E_n has $\{A : A \subseteq \{1, \cdots, 2n-1\}; |A| \leq n-1\}$ as vertex-set, and two vertices A and B are joined by an edge whenever $|A \Delta B| = 1$ or $|A \Delta B| = 2n - 2$.

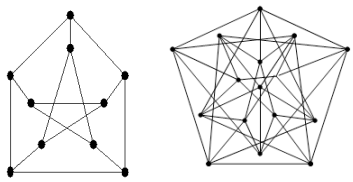


Fig. 3. Petersen Graph Greenwood Gleason Graph

2 Preliminary Results About the Cycle Regular Graphs

H.M.Mulder [9] introduced (0,λ)-graphs for which each two distinct vertices have 0 or λ common neighbours and proved that maximum (0,λ)-graphs are hypercubes. One way of generalization of this concept is to consider cycle regular graphs which have also some regularity properties and that maximum for a particular case is related to hypercubes .

Definition 1. *(Mollard [8]) A graph $G = (V, E)$ of girth at least μ ($\mu \geq 2$) is a $[\mu, \lambda]$- cycle regular graph ($\lambda \geq 1$), if there is a non empty subset C of elementary cycles in G such as every path $P_{\mu+1}$ in G belongs to exactly λ cycles in C. In the particular case, where C is the set of elementary cycles of a given length η ($\eta \geq 2\mu$), we say that G is a $[\mu, \lambda, \eta]$*-cycle regular graph *(also called a cycle regular graph).*

We can say now that(0,λ)-graphs are the $[2, \lambda - 1, 4]$-*cycle regular graphs.* Our study is focuses on the [3,1,6]- *cycle regular graphs*, like the odd graphs O_n ($n > 2$). The [3,1,6]-cycle regular graphs are triangle-free, since the triangle cannot belong to an elementary cycle of length greater than three. So, for any vertex u, the set $N(u)$ is stable. In this class, we can consider also the subgraph L_n^k.

Mulder [9] and Laborde and Rao Hebbare [6] have shown separately, that for a given degree and of all the $[2, 1, 4]$-cycle regular graphs, the hypercube is of maximum order. On the other hand, Mollard [7] showed that, for a given degree, the hypercube is of maximum diameter among these graphs. Furthermore, for a given degree n, he showed that H_n is of maximum order among the $[3, 1, 6]$-cycle regular graphs [8]. In this paper, we give some new characterizations of H_n in the class of graphs which are $[3, 1, 6]$-cycle regular graphs. Moreover, we give other properties for these latest graphs.

Proposition 1. *(Mollard [8]) Let $u_1u_2 \cdots u_\mu$ be a path of length $\mu-1$ of a $[\mu, \lambda]$-cycle regular graph G. Then $d(u_1) = d(u_\mu)$.*

Proposition 2. *(Mollard [8]) Every vertex of a $[\mu, \lambda]$-cycle regular graph is at least on one cycle in a subset C of elementary cycles.*

Proposition 3. *(Mollard [8]) If G is a $[\mu, \lambda]$-cycle regular graph of minimal degree $\delta(G) \geq 3$, then G is regular or semi-regular.*

3 The [3,1,6]-Cycle Regular Graph

Mollard [8] gave an upper bound for the order of a [3,1,6]-cycle regular graph, for a given degree. Moreover, he gave a characterization of the subgraph H_n.

Proposition 4. *(Mollard [8]) Let G be a [3,1,6]-cycle regular graph of maximum degree n. Then*

1. $|V(G)| \leq \binom{2n}{n}$,
2. $|V(G)| = \binom{2n}{n}$ *if and only if G is H_n.*

For a given degree, we show that H_n is of maximum diameter among the [3,1,6]-cycle regular graphs.

Theorem 1. *Let G be a [3,1,6]-cycle regular graph of maximum degree $n \geq 2$. Then*

1. $diam(G) \leq 2n - 1$
2. $diam(G) = 2n - 1$ *if and only if G is H_n.*

To establish the proof of Proposition 4, Mollard used these two following Propositions 5 and 6, which we also use to prove Theorem 1.

Proposition 5. *(Mollard [8]) Let G be a [3,1,6]-cycle regular graph and for an arbitrary level decomposition $\{N_0, N_1, \ldots, N_p\}$ of G, let $u \in N_i$. Then $d^-(u) = m(u, N_{i-1}) \geq \lceil \frac{i}{2} \rceil$.*

Proposition 6. *(Mollard [8]). Let G be a [3,1,6]-cycle regular graph of maximum degree n and $\{N_0, N_1, \cdots, N_p\}$ a level decomposition from a vertex of degree n. Then, for $k = 0, \cdots, n-2$*

$$|N_{2k+1}| \leq \frac{n}{k+1} \left(\binom{n-1}{k}\right)^2 \tag{1}$$

and

$$|N_{2k+2}| \leq \frac{n(n-k-1)}{(k+1)^2} \left(\binom{n-1}{k}\right)^2 \tag{2}$$

Proof. (of Theorem 1)

1. From a vertex u of a [3,1,6]-cycle regular graph G, let $\{N_0, N_1, \cdots, N_p\}$ the corresponding level decomposition . By Proposition 5, $p \leq 2n - 1$.
2. The [3,1,6]-cycle regular graph H_n is a graph of degree n and diameter $2n-1$. Now, let G be a [3,1,6]-cycle regular graph of maximum degree n and diameter $2n - 1$. Without loss of generality, consider a level decomposition of G from a vertex of degree n. Suppose that G is not isomorphic to H_n. According to Proposition 4,

   $$|V(G)| < \binom{2n}{n} \tag{3}$$

   By counting the edges between N_{2n-1} and N_{2n-2}, we obtain $n|N_{2n-1}| \leq |N_{2n-2}|$, but $|N_{2n-2}| \leq n$.

Since $diam(G) = 2n-1$, so $|N_{2n-1}| = 1$
According to Proposition 6 and the inequality (3), there is an index k, $0 \leq k \leq n-2$, such as either

$$|N_{2k+1}| < \frac{n}{k+1}\left(\binom{n-1}{k}\right)^2 \tag{4}$$

or

$$|N_{2k+2}| < \frac{n(n-k-1)}{(k+1)^2}\left(\binom{n-1}{k}\right)^2 \tag{5}$$

Suppose that (4) is valid. Proposition 5 indicates that:
$\forall u \in N_{2k+2}$: m$(u,N_{2k+1}) \geq k+1$.
By counting the edges between N_{2k+1} and N_{2k+2}, we obtain
$|N_{2k+2}| \leq \frac{n-k-1}{k+1}|N_{2k+1}|$.
After using the expression (4), we get:
$|N_{2k+2}| < \frac{n(n-k-1)}{(k+1)^2}\left(\binom{n-1}{k}\right)^2$.
By iteration, we obtain:
$\forall h \geq k$, $|N_{2h+1}| < \frac{n}{h+1}\left(\binom{n-1}{h}\right)^2$ and $|N_{2h+2}| < \frac{n(n-h-1)}{(h+1)^2}\left(\binom{n-1}{h}\right)^2$. On the one hand, for $u \in N_{2n-1}$: $m(u, N_{2n-2}) \geq \lceil \frac{2n-1}{2} \rceil = n$. On the other hand, $|N_{2n-2}| < n$, which is absurd. □

Theorem 2. *Let G be a bipartite [3,1,6]-cycle regular graph of maximum degree n. Then, G is regular if and only if G is H_n.*

Proof. It is obvious that H_n is a bipartite regular [3,1,6]-cycle regular graph of degree n.

Let G be a regular bipartite [3,1,6]-cycle regular graph of degree n and assume that G is not H_n. From Proposition 4 and Theorem 1, $|V(G)| < \binom{2n}{n}$ and $diam(G) < 2n-1$. So for $\{N_0, ..., N_p\}$ a level decomposition of G, there is $k \in \{0, 1, ..., p\}$, such that
$\forall h \geq k$, $|N_{2h+1}| < \frac{n}{h+1}\left(\binom{n-1}{h}\right)^2$ and $|N_{2h+2}| < \frac{n(n-h-1)}{(h+1)^2}\left(\binom{n-1}{h}\right)^2$.
On the one hand, for $u \in N_p$: $m(u, N_{p-1}) = n \geq \lceil \frac{p}{2} \rceil$. On the other hand, $|N_{p-1}| < n$, which is absurd. □

Proposition 7. *Let u and v be two vertices having at least two common neighbours, in a [3,1,6]-cycle regular graph G, with $\delta(G) \geq 2$. Then u, v and $N(u) \cap N(v)$ are on the same 6-cycle.*

Proof. Let u and v be *two* vertices of a [3,1,6]-cycle regular graph, having at least *two* common neighbours, denoted a and b. Then the path v, a, u, b belongs to a single 6-cycle $\beta = v, a, u, b, c, d, v$. If there is $e \in N(u) \cap N(v)$ not on β, the path d, c, b, u would belong to at least *two* 6-cycles: β and d, c, b, u, e, v, d, absurd. □

Corollary 1. *Let $A = \{x, y, z, t, u, v\}$ be the vertex set of one 6-cycle, in a [3,1,6]-cycle regular graph G, such as d(x,t) = 1. Then the subgraph G_A induced by A is isomorphic to $K_{3,3}$.*

Proof. Let G be a [3,1,6]-cycle regular graph and A be the vertex set of a 6-cycle $\beta = x, y, z, t, u, v, x$, such as $d(x, t) = 1$. The path z, t, x, v belongs to a single 6-cycle z, t, x, v, x', t', z. If $x' \notin A$ and $t' \notin A$; the path z, y, x, v belongs to at least *two* 6-cycles in $G : z, y, x, v, x', t', z$ and β absurd. The cases ($x' = u$ and $t' \notin A$) and ($x' \notin A$ and $t' = y$) do not take place, according to the theorem 2. Then $x' = u$ and $t' = y$. By symmetry, we can conclude that $zv \in E(G)$. □

Proposition 8. *Let G be a [3,1,6]-cycle regular graph of maximum degree n≥ 2. If every two vertices have exactly three common neighbours then G is $K_{3,3}$.*

Proof. Let G be a [3,1,6]-cycle regular graph of maximum degree n≥ 2. If every *two* vertices have exactly *three* common neighbours, so each pair of vertices are of distance *one* or *two*. Hence, the $diam(G) = 2$ and G would be regular of degree 3. It is the bipartite graph $K_{3,3}$. □

Theorem 3. *Let G be a [3,1,6]-cycle regular graph, with $\delta(G) \geq 2$. Then $\forall u, v \in V(G)$: $|N(u) \cap N(v)| \in \{0, 1, 3\}$.*

Proof. Let u, v be *two* vertices of G having a as common neighbour. As $\delta(G) \geq 2$, there is $b \in N(u)$. b is distinct from v, since G is a [3,1,6]-cycle regular graph. The path v, a, u, b belongs to a single 6-cycle $\beta = v, a, u, b, c, d, v$. According to the theorem 2, if u and v admitted an other common neighbour, it would be b or d. Without loss generality, suppose that $b \in (N(u) \cap N(v)) \setminus \{a\}$. According to the corollary 2, d will be the third common neighbour of u and v. Thus, $N(u) \cap N(v) = \{a, b, d\}$. From where the result. □

4 An Operation on the [3,1,6]-Cycle Regular Graphs

We give an operation which is stable in this class

Proposition 9. *If G and H are two [3,1,6]-cycle regular graphs, then $G \times H$ is a [3,1,6]-cycle regular graph.*

Proof. Let G and H be two [3,1,6]-cycle regular graphs. Let $G \times H$ be the categorical graph of G and H, of vertex set $V(G \times H) = V_1 \times V_2$, with $V_1 = V(G)$ and $V_2 = V(H)$. Let $I = xyzt$ be P_4 in $G \times H$, where $x = (x_1, x_2)$, $y = (y_1, y_2)$, $z = (z_1, z_2)$ and $t = (t_1, t_2)$. By construction of $G \times H$, x_1y_1, y_1z_1 and z_1t_1 (resp. x_2y_2, y_2z_2 and z_2t_2) are in $E(G)$ (resp. $E(H)$). As G (resp. H) is a [3,1,6]-cycle regular graph, there are two vertices u_1 and v_1 (resp. u_2 and v_2) in G (resp. H) such that x_1v_1, v_1u_1, u_1t_1 (resp. x_2v_2, v_2u_2, u_2t_2) are in $E(G)$ (resp. $E(H)$), forming the 6-cycle $\beta_1 = x_1y_1z_1t_1u_1v_1x_1$ (resp. $\beta_2 = x_2y_2z_2t_2u_2v_2x_2$). Then, there is in $G \times H$ a 6-cycle $\beta = xyztuvx$, which contains I, where $u = (u_1, u_2)$ and $v = (v_1, v_2)$. The 6-cycle β is the single 6-cycle in $G \times H$ containing I. Else, there are two vertices $a = (a_1, a_2)$ and $b = (b_1, b_2)$ in $G \times H$, such as : xb, ba, at are in $E(G \times H)$ (a (resp. b) can be possibly u (resp. v) but not both $a = u$ and $b = v$). Then, there is a second 6-cycle $\rho_1 = x_1y_1z_1t_1a_1b_1x_1$ (resp. $\rho_2 = x_2y_2z_2t_2a_2b_2x_2$) in G (resp. H) distinct from β_1 (resp.β_2), which is absurd. □

Corollary 2. . *$G \times K_2$ is a [3,1,6]-cycle regular graph if and only if G is a [3,1,6]-cycle regular graph.*

Proof. By Proposition 10, we have only to prove the efficiency. Assume that $G \times K_2$ is a [3,1,6]-cycle regular graph and set $V(K_2) = \{0, 1\}$. Consider a path P_4: $I' = x'y'z't'$ in G. Without loss of generality, consider the path P_4: $I = xyzt$, with $x = (x', 1)$, $y = (y', 0)$, $z = (z', 1)$ and $t = (t', 0)$. As $G \times H$ is a [3,1,6]-cycle regular graph, I belongs to the single 6-cycle $\beta = xyztuvx$, where $u = (u', 1)$ and $v = (v', 0)$. So the path $I' = x'y'z't'$ will belong to the 6-cycle $\beta' = x'y'z't'u'v'x'$. The cycle β' is the single 6-cycle which contains I'; otherwise denote by $\rho' = x'y'z't'a'b'x'$ (a'(resp. b') can be u'(resp. v') but not both a'=u' and b'=v') an other 6-cycle containing I'. Hence, there is a seconde 6-cycle $\rho = xyztabx$, in $G \times H$, distinct from β and containing I, which is absurd. □

Lemma 1. *Let u and v be two vertices having at least two common neighbours, in a [3,1,6]-cycle regular graph G, with $\delta(G) \geq 2$. Then u, v and $N(u) \cap N(v)$ are on the same 6-cycle.*

Proof. Let u and v be two vertices of a [3,1,6]-cycle regular graph, having at least two common neighbours, denoted by a and b. Then the path $vaub$ belongs to a single 6-cycle $\beta = vaubcdv$. If there is $e \in N(u) \cap N(v)$ not on β, the path $dcbu$ will belong to at least two 6-cycles: β and $dcbuevd$, which is absurd. □

Corollary 3. *Let $\beta = xyztuvx$ be a 6-cycle in a [3,1,6]-cycle regular graph G, such that $xt \in E$. Then the subgraph induced by the vertices of β is isomorphic to $K_{3,3}$.*

Proof. Let G be a [3,1,6]-cycle regular graph and A be the vertex set of a 6-cycle $\beta = xyztuvx$, such that $xt \in E$. The path $ztxv$ belongs to a single 6-cycle $ztxvx't'z$. If $x' \notin A$ and $t' \notin A$; the path $zyxv$ belongs to at least two 6-cycles in $G : zyxvx't'z$ and β, which is absurd. The cases ($x' = u$ and $t' \notin A$) and ($x' \notin A$ and $t' = y$) do not occur, according to Lemma 1. Then $x' = u$ and $t' = y$. By symmetry, we can deduce that $zv \in E(G)$. □

References

[1] Bondy,J.A., Murty,U.S.R.: Graph Theory with applications, Macmillan & Co, London 1976.

[2] Havel,I.: Semipaths in directed cubes, in: Graphs and other Combinatorial Topics, **B. 59**(Teubneur Texte zumMathematik, Teubneur,Leipzig) (3° Symposium Tchescoslovaque de Th. Graphs), (1982).

[3] Berrachedi,A., Kahoul,N. : Graphs with special induced 6-cycle. Actes du colloque sur l'optimisation et les systèmes d'information *COSI'04*, Tizi-ouzou, Algérie (7-9 juin 2004).

[4] Berrachedi,A., Kahoul,N.: Cyclic Regularity in Some Particular Graphs. Acte du colloque sur l'optimisation et les sytmes d'information *COSI'05*, Bejaïa, Algérie (12-14 juin 2005).

[5] Berrachedi,A., Kahoul,N.: Special Cyclic Construction in Some Particular Graphs. Colloque International sur l'optimisation et les systmes d'information *COSI'06*, Alger-Algérie (11-13 juin 2006).

[6] Laborde,J.M., Rao Hebbare,S.P.: Another chararacterization of hypercubes. Discrete. Math, **39**(82) 161–166

[7] Mollard,M.: Les invariants du n-cube, Thèse 3ème cycle, Université Joseph Fourier, Grenoble 1981.

[8] Mollard,M.: Quelques problèmes combinatoires sur l'Hypercube et les graphes de Hamming, Thèse Doctorat es-Science, Université Joseph Fourier, Grenoble 1989.

[9] Mulder,H.M.: $(0,\lambda)$-graphs and n-cubes, Discrete . Math, **28** (1979) 179–188.

[10] Mulder,H.M.: The interval function of a graph, Mathematics Centre Tracts 132, Mathematisch Centrum, Amesterdam 1980.

Analyse Discriminante Linéaire Bidimensionnelle Neuronale pour l'Identification

Chahrazed Rouabhia et HichamTebbikh

Laboratoire d'Automatique et Informatique de Guelma -LAIG-
Université 8 mai 45 de Guelma (Algérie)
c_rouabhia@yahoo.fr, tebbikh@yahoo.com

Résumé. Cet article propose une approche hybride d'identification faciale qui consiste à entraîner un perceptron multicouche (PMC) avec les caractéristiques de personnes obtenues par l'analyse discriminante linéaire bidimensionnelle orientée en colonnes (ADL2DoC). Notre approche se présente en deux versions: l'une globale appliquée sur le visage entier et l'autre locale appliquée sur deux régions faciales. En utilisant la base de visages du LAIG qui contient 40 personnes de sexes différents, chacun ayant une séquence de 10 images, nous avons pu conclure que la version locale est la plus performante. Neuro-ADL2DoC locale bénéficie donc de la fusion combinant à la fois le pouvoir discriminant de l'ADL2D, le pouvoir de généralisation du PMC et la discrimination des éléments faciaux pris séparément.

Mots clés: ADL2D, PMC, yeux, nez, séquences vidéo.

1 Introduction

La biométrie est une technologie qui vise la reconnaissance de personnes en se basant sur la connaissance de: i) leurs *caractéristiques physiologiques* (visage, empreintes digitales, iris, géométrie de la main, géométrie de l'oreille, etc.), ii) leurs *caractéristiques comportementales* (signature manuscrite, démarche, rythme de frappe sur un clavier d'ordinateur, etc.) ou iii) leurs *caractéristiques biologiques* (ADN, odeur corporelle, salive, etc.). Les systèmes biométriques effectuent deux types de comparaison à savoir:

- *un à plusieurs:* pour répondre à la question « qui suis-je ? »; ces systèmes visent à établir l'identité d'une personne en comparant ses caractéristiques avec celles de la base de données, et
- *un à un:* pour effectuer une vérification ou authentification et répondre à la question « suis-je effectivement la personne que je déclare ? » et cela en comparant un gabarit de vérification avec un autre d'enrôlement.

Dans le cadre de notre recherche, nous intéressons aux systèmes d'identification qui sont des systèmes typiques de reconnaissance de formes. Pour la phase d'extraction de caractéristiques et de réduction de dimensions, l'analyse en composantes principales (ACP) [1] et l'analyse discriminante linéaire (ADL) [2][3] sont les plus

utilisées. Quant à la phase de classification, la règle du plus proche voisin est la plus populaire utilisant différentes mesures de similarité à savoir: la distance euclidienne classique [1][2] et la distance euclidienne pondérée [3], la distance cosinus [4] et la distance city-block [5]. Récemment, une variante de ce classificateur, reposant sur le calcul de la distance euclidienne entre deux matrices au lieu des vecteurs, a été proposée, en 2004, par Yang dans [6] et adoptée ensuite par [7][8].
Nous intéressons particulièrement à l'ADL qui pésente les avantages suivants [3]: i) elle possède un pouvoir discriminant qui dépasse celui de l'ACP, ii) elle est un bon classificateur pour les données linéairement séparables, et iii) elle est facile à implémenter. Malgré ses avantages, elle souffre du problème de la singularité de la matrice de covariance intra-classe du au fait que les vecteurs de visages sont de tailles très grandes comparées au nombre de visages disponibles pour l'apprentissage. Pour corriger ce problème, Swets et Weng [9] et Belhumeur et al. [2] ont proposé de précéder l'ADL d'une ACP, quant à Zhao et al. [3], ils ont modifié la matrice de covariance intra-classe. D'autres méthodes ont été, également, proposées parmi lesquelles nous comptons l'ADL Pseudo-Inverse [10], Regularized LDA [11] et Direct-LDA [12]. Mais récemment, de nouvelles approches utilisant une représentation bidimensionnelle ont été introduites à savoir l'ADL bidimensionnelle orientée en ligne et orientée en colonnes (ADL2DoL et ADL2DoC) [7], l'ADL2D bilatérale [8] et l'Analyse Discriminante Bilinéaire (ADB) [13].

A notre connaissance, il n'y a pas un système utilisant une méthode bidimensionnelle pour l'extraction de caractéristiques et un PMC pour la classification à partir de séquences vidéo, c'est pourquoi, nous avons pensé à faire cette fusion et l'appliquer :

- sur des visages entier après leur localisation automatique, et
- sur des régions locales contenant les yeux avec les sourcils et le nez après la localisation de ces éléments faciaux.

dans le but de comparer la discrimination des éléments faciaux pris séparément avec le visage entier.

La 2ème partie de cet article présente donc le système d'identification proposé Neuro-ADL2DoC en détaillant ses différentes étapes et les résultats obtenus, quant à la 3ème partie, elle est dévolue à présenter Neuro-ADL2DoC locale au sens où elle s'applique sur les régions faciales au lieu de tout le visage. Ses résultats de simulations seront également discutés et enfin nous concluons.

2 Identification par Neuro-ADL2DoC globale

L'idée du système proposé (figure 1) qui comporte trois étapes: i) *localisation automatique de visages*, ii) *extraction de caractéristiques* par l'ADL2DoC et iii) *classification* par un PMC consiste à réduire les redondances au niveau des colonnes des matrices de visages et entraîner un PMC par les caractéristiques obtenues pour retourner l'identité des personnes à reconnaître.

Analyse Discriminante Linéaire Bidimensionnelle Neuronale pour l'Identification

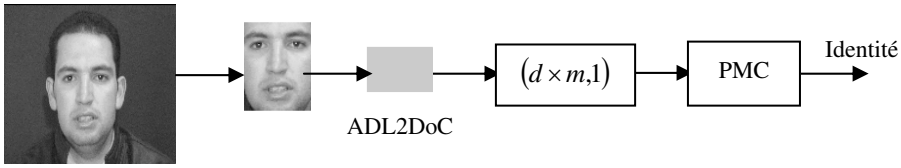


Fig. 1. Système Neuro-ADL2DoC.

Pour nos tests, la base de visages utilisée a été acquise au LAIG en 2003/2004 sous des conditions d'éclairage contrôlées et avec des prises de vues frontales et presque frontales. Cette base contient 400 images de 40 personnes (20 hommes et 20 femmes) ayant chacun une séquence de 10 images. Lors de l'acquisition des images, chaque personne est invitée à prononcer la phrase «reconnaissance automatique de visages ».

Fig. 2. Extrait de la base de visages B-LAIG.

Souvent, la segmentation de visages à des fins de reconnaissance est effectuée manuellement [2] ou après la localisation des yeux [3]. En se basant sur la méthode de [14] pour détecter les régions des yeux, nous avons localisé automatiquement dans [15] les visages de la base B-LAIG (voir figure 1). Cette première étape permet d'éliminer les arrières plans, les cheveux et les voiles qui peuvent influencer la reconnaissance et retournent de nouvelles images de dimensions (73×56). Ces dernières ne contiennent que les visages et assurent ainsi une identification basée uniquement sur la forme des visages segmentés.

2.1 Extraction de caractéristiques par l'ADL2D orientée en colonnes

Récemment, une méthode bidimensionnelle reposant sur l'analyse discriminante linéaire (ADL) a été introduite indépendamment par Visani et al. [7] et Kong et al.[8]. Elle est dite Analyse Discriminante Linéaire Bidimensionnelle orientée (ADL2Do) dans [7] et Analyse Discriminante Bidimensionnelle de Fisher (ADF2D) dans [8]. Cette méthode qui est l'ADL appliquée séparément sur les colonnes ou les lignes des matrices [8] se présente en deux versions; ADL2DoC (orientée en colonnes) et ADL2DoL (orientée en lignes) ou LU2DFDA et RU2DFDA (Left-multiplying Unilateral et Right-multiplying Unilateral 2D Fisher Discriminant Analysis respectivement). Au contraire des méthodes globales [1][2][3][4] qui transforment les

matrices de visages en vecteurs, l'ADL2D repose sur l'utilisation directe des matrices de visages et présente les avantages suivants : i) elle est plus performante que l'ACP2D et Fisherfaces [7], ii) elle réduit le temps de calcul de la matrice de covariance et iii) corrige le problème de la singularité ce qui la rend applicable pour les problèmes de petites tailles (Small Sample Size problem ou SSS).

2.2 Classification par un perceptron multicouche

L'application de l'ADL2DoC sur l'ensemble d'apprentissage, formé de matrices de visages de tailles $(n \times m)$, donne la matrice de projection et les matrices caractéristiques. Ces dernières doivent être transformées en vecteurs avant de les fournir au PMC qui contient 3 couches : i) la 1ère présente les exemples au réseau et comporte autant de neurones que de caractéristiques, ii) une couche cachée, au sens où ses sorties ne sont pas directement observables, le nombre de ses neurones est déterminé empiriquement et iii) une couche de sortie qui fournit à l'extérieur les réponses du réseau, elle contient 40 neurones correspondant au nombre de personnes de la base de visages utilisée. Au cours de l'apprentissage, le PMC utilisé a été entraîné par l'algorithme populaire «Backpropagation» et comme ses performances dépendent fortement de l'initialisation des poids, du pas d'apprentissage et du nombre de neurones de la couche cachée; nous avons réalisé plusieurs tests avant de préserver la topologie optimale (Pour plus de détails sur le PMC voir [16]). Quant à la phase de test, elle consiste à projeter la matrice de chaque visage à reconnaître dans le nouveau sous espace via la matrice de projection déterminée dans la phase d'apprentissage et effectuer ensuite la classification.

2.3 Résultats expérimentaux

L'évaluation des résultats obtenus par Neuro-ADL2DoC globale est effectuée en partitionnant aléatoirement la base de visages en deux sous bases de 200 images chacune, l'une pour l'apprentissage et l'autre pour le test. Les taux d'identification reportés (Tableau 1) pour chaque cas de vecteurs propres retenus « d » sont la moyenne de six partitions de test. Le meilleur résultat (97.25%) est obtenu pour 6 vecteurs propres correspondant à une image caractéristique de taille (6×56).

Tableau 1. Taux de reconnaissance (%) en fonction de différents nombres de vecteurs propres.

Neuro-ADL2DoC						
d × m	2×56	4×56	6×56	8×56	10×56	12×56
Taux (%)	95.66	96.75	**97.25**	96.91	97.00	96.66

La figure 4 illustre l'évolution du taux de reconnaissance et de l'erreur quadratique moyenne obtenus sur la base d'apprentissage.

Analyse Discriminante Linéaire Bidimensionnelle Neuronale pour l'Identification

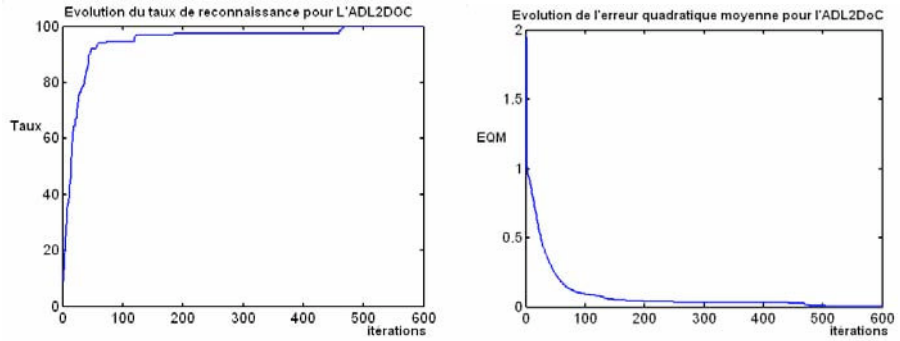


Fig. 4. Evolution du taux de reconnaissance (à gauche) et de l'erreur quadratique moyenne (à droite) obtenus sur l'ensemble d'apprentissage pour l'ADL2DoC ($d = 10$) .

Le résultat obtenu dépasse également les résultats des approches Neuro-ACP2D et Neuro-ACPDL2D que nous avons proposé dans [15] comme le montre le tableau 2.

Tableau 2. Comparaison des résultats des approches globales bidimensionnelles.

Méthodes	**Neuro-ADL2DoC**	**Neuro-ACPDL2D [15]**	**Neuro-ACP2D [15]**
Taux (%)	**97.25**	97.00	96.50

3 Identification par Neuro-ADL2DoC Locale

Le besoin de reconnaître des personnes dont les éléments faciaux sont partiellement occultés soit par le port de lunettes, d'un cache-nez, ou même par une barbe et/ou une moustache, etc., le pouvoir discriminant de ces éléments et la performance de l'ADL2D [7] nous ont incité à proposer le système de la figure 5. Il consiste, après une localisation automatique des sourcils, des yeux et du nez, à appliquer l'ADL2DoC [7] sur deux régions faciales et faire ensuite la classification par un PMC d'où le nom Neuro-ADL2DoC locale.

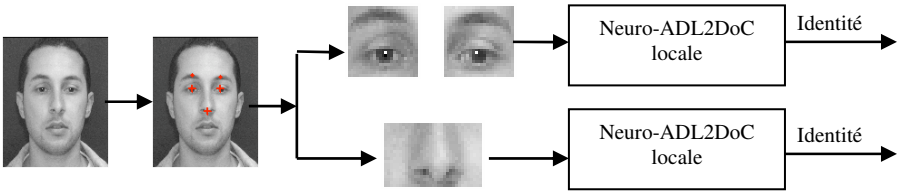


Fig. 5. Principe de l'identification par Neuro-ADL2DoC locale.

3.1 Localisation des éléments faciaux et extraction de régions locales

En raison de la prise de vues frontales et presque frontales des images des séquences, et de la source d'éclairage, aussi frontale sur les iris, la méthode adoptée pour localiser les yeux est celle proposée dans [17]. Quant à la localisation du nez et de la bouche à partir des images originales, la méthode de [17] n'a pas donné de correctes localisations notamment pour les hommes moustachus et les femmes portant un voile. C'est pourquoi, nous avons proposé dans [14] la fusion de la méthode de [17] avec la segmentation du visage en 3 parties tout en utilisant la géométrie du visage.
L'extraction de régions contenant les yeux et le nez revient à extraire deux rectangles de tailles suffisantes pour ne contenir que le sourcil et l'œil gauche et le sourcil et l'œil droite chacun (Figure 6). Pour les régions des nez, nous avons veillé à ce qu'elles ne contiennent que les nez sans les moustaches, pour assurer une correcte identification basée uniquement sur la forme du nez compte tenu que les hommes peuvent laisser ou enlever leur moustaches. Quant à la bouche, sa région a été éliminée à cause de la présence d'importantes variations intra-classes dues à l'activité labiale durant l'acquisition des images comme le montre la figure 6.

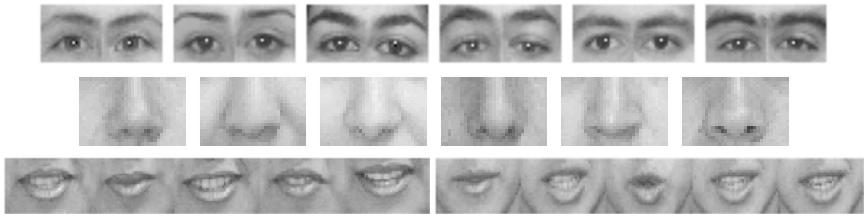


Fig. 6. Exemples de régions contenant les éléments faciaux (3ème ligne : mouvements des lèvres de deux personnes).

3.2 Résultats expérimentaux

Le tableau 3 donne les résultats de Neuro-ADL2DoC appliquée sur les régions des yeux (Yeux-ADL2DoC) et sur celle des nez (Nez-ADL2DoC) en fonction de différents nombres de vecteurs propres retenus. Pour chaque cas et chaque méthode, les taux reportés sont la moyenne de six partitions aléatoires de test.

Tableau 3. Taux de reconnaissance en fonction de différents nombres de vecteurs propres.

Yeux-ADL2DoC						
m× d	2×46	4×46	6×46	8×46	10×46	12×46
Taux (%)	97.66	98.83	98.75	**99.25**	99.16	99.25
Nez-ADL2DoC						
m× d	2×33	4×33	6×33	8×33	10×33	12×33
Taux (%)	97.16	**97.75**	97.25	97.08	97.00	97.16

Analyse Discriminante Linéaire Bidimensionnelle Neuronale pour l'Identification

Nous avons comparé nos résultats avec l'approche géométrique Neuro-ACP proposée dans [18] et testée également sur la base B-LAIG avec un PMC. Le tableau 4 montre clairement la performance de Neuro-ADL2DoC globale et locale vis-à-vis Neuro-ACP.

Tableau 4. Comparaison des résultats (taux en %).

Neuro-ADL2DoC Locale				**Neuro-ADL2DoC Globale**		**Neuro-ACP [18] Géométrique**
Yeux		**Nez**		**Visage entier**		**8 distances**
Taille	Taux	Taille	Taux	Taille	Taux	Taux
8×46	99.25	4×33	97.75	6×56	97.25	94.50

Conclusion

Par le système d'identification proposé Neuro-ADL2DoC qui fusionne l'ADL2DoC pour l'extraction de caractéristiques et le perceptron multicouche pour la classification, nous avons étudié les performances d'identification de personnes par les yeux et le nez séparément vis-à-vis le visage entier et comparé nos résultats avec l'approche géométrique de [18]. Les régions des yeux (avec les sourcils) sont les plus discriminantes (99.25%) suivies de la région du nez (97.75%). Ces dernières dépassent, en termes d'identification, le visage entier et les distances géométriques également appliquées avec un PMC. Cela est du à la performance élevée de l'ADL2D [7], à la généralisation du PMC et aux formes spécifiques des sourcils, des yeux et du nez à chaque personne. Une prochaine étape de notre travail portera sur l'amélioration de l'approche géométrique.

Remerciements. Les auteurs remercient infiniment Mr.Boualleg Abdelhalim pour l'utilisation de sa base de visages B-LAIG.

Références

1. Turk, M.A., Pentland, A.P.: Eigenfaces for Recognition. Journal of Cognitive Neuroscience, (1991)3(1):71-86
2. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. Fisherfaces: Recognition using Class Specific Linear Projection. In IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.19, No.7 (1997) 711-720
3. Zhao, W., Chellappa, R., Krishnaswamy, A.: Discriminant Analysis of Principal Components for Face Recognition. In Proceedings, International Conference on Automatic Face and Gesture Recognition (1998) 336-341
4. Bartlett, M.S., Lades, H.M., Sejnowski, T.: Independent Component Representation for Face Recognition. In Proceedidings, SPIE Symposium on Electronic Imaging: Science and Technology. (1998) 528-539

5. Laskri, M.T., Chefrour, D.: Who-Is : Système d'Identification des Visages Humains. Arima, Vol.1 (2002) 39-61
6. Yang, J. Zhang, D., Frangi, A.F., Yang, J.: Two Dimensional PCA: A New Approach to Appearance-Based Face Representation and Recognition. IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol.26, No.1 (2004)
7. Visani, M. Garcia, C., Jolion, J.M.: Two Dimensional-Oriented Discriminant Analysis for Face Recognition. In Proc.of the Int. Conf. On Computer Vision and Graphics ICCVG'04 à paraître dans la série Computational Imaging and Vision, Varsovie, Pologne (2004)
8. Kong, H., Wang, L., Teoh, E.K., Wang, J-G., Venkateswarlu, R.: A Frame Work of 2D Fisher Discriminant Analysis: Application to Face Recognition With Small Number of Training Samples. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, USA (2005)1083-1088
9. Swets,D.L., Weng,J.: Using Discriminant Eigenfeatures for Image Retrieval. IEEE Trans.on PAMI, Vol.18, (1996) 831-836
10. Raudys S., Duin, R.P.W.: On Expected Classification Error of the Fisher Linear Classifier with Pseudo-Inverse Covariance Matrix. Pattern Recognition Letter (1998)
11. Lu, J., Plataniotis, K.N., Venetsanopoulos, A.N.: Regularized Discriminant Analysis for the Small Sample Size Problem in Face Recognition. Pattern Recognition Letter (2003)
12. Yu, H., Yang, J.: A Direct LDA Algorithm for High-Dimensional Data with Application to Face Recognition. Pattern Recognition (2001)
13. Visani, M., Garcia, C., M.Jolion, J.: Face Recognition using Modular Billinear Discriminant Analysis. Dans Proceedings of the International Conference on Visual Information Systems (VIS 2005), Amsterdam, Pays-Bas (2005) 24-34
14. Peng, K., Cheng, L.: A Robust Algorithm for Eye Detection On Gray Intensity Face Without Spectacles. JCS &T.Vol.5, No.3 (2005) 127-132
15. Rouabhia, Ch., Boualleg, A/h., Tebbikh, H.:Neuro-ACPDL2D Globale et Locale: Approches Hybrides Bidimensionnelles pour la Reconnaissance Automatique de Visages. Actes de la 5ème conférence sur le Génie Electrique (CGE'05), Bordj El Bahri, Alger, Algérie, 16-17 avril 2007
16. Sima, J.: Introduction to Neural Networks. Technical Report. N. V-755 (1998)
17. Leroy, B., Chouakria, A., Herlin, L., Diday E. : Approche Géométrique et Classification pour la Reconnaissance de Visage. INRIA, France (1996)
18. Boualleg, A/H., Bencheriet, Ch., Tebbikh, H. : Approche Géométrique pour la Reconnaissance Automatique de Visages par une Approche Neuro-ACP. International Symposium on Signal and its Application (ISSPA'2005), Mostaghanem (2005)

Solving MAX-SAT Problems using a Scatter Search Metaheuristic

Dalila Boughaci [1, 2], Belaïd Benhamou [1] and Habiba Drias[2]

[1] INCA/LSIS - UMR CNRS 6168, CMI 39 rue Frédric Joliot-Curie
13453 Marseille cedex 13
boughaci@cmi.univ-mrs.fr

[2] LRIA / USTHB
Université des Sciences et de la Technologie Houari- Boumediène
BP 32 El-Alia, Beb-Ezzoaur, 16111, Alger, Algérie.
dboughaci@usthb.dz

Abstract. In this work, a scatter search metaheuristic is studied for the satisfiability problem, in particular for its optimization version namely MAX-SAT. The paper describes a scatter search algorithm enhanced with a stochastic local search (SLS) component combined with a specific combination method. The latter is used to identify promising search regions while the stochastic local search performs an intensified search of solutions around these regions. Our objective is to achieve a good compromise between intensification and diversification in the search process. Experiments on random problems instances and several instances from MAX-SAT libraries are performed to show and compare the effectiveness of our approach.

1 Introduction

The Boolean satisfiability problem (SAT) is of central importance in various areas of computer science, including theoretical computer science, algorithmic, artificial intelligence, hardware design and verification. Formally, given a collection of m clauses[1] $C=\{C_1, C_2,.. C_m\}$ involving n Boolean variables $\{X_1, X_2,.. X_n\}$, the SAT problem is to decide whether an assignment of values to variables exists such that all the clauses are simultaneously satisfied. The maximum satisfiability problem (MAX-SAT) is the optimization variant of SAT. Given a propositional formula in conjunctive normal form (CNF), the MAX-SAT problem is to find a variable assignment that maximizes the number of satisfied clauses. MAX-SAT is NP-Hard [5] even when each clause has no more than two literals, while SAT with two literals per clause can be solved in polynomial time.

Many methods for SAT and MAX-SAT have been proposed. These methods can be divided into three categories: the exact methods, the stochastic local search (SLS) methods and the evolutionary methods.

[1] A clause is a disjunction of literals. A literal is a variable or its negation.

a) The exact methods are dedicated to solve the decision version of SAT problems. The well-known exact algorithms are based on the Davis-Putnam-Loveland procedure [4]. Among the exact algorithms for MAX-SAT, we cite [1, 3, 10, 25].

b) The local search methods for SAT problems have been researched intensively since the introduction of GSAT [23], the famous local search method which tries to maximize the number of satisfied clauses by a greedy ascent in the space of variable assignments. Many good ideas have been proposed to enhance the local search method such as the noise strategy [24], tabu search [19], simulated annealing [11], guided local search [21], Novelty [20], adaptNovelty [13] and recently G2wSAT[17].

c) The evolutionary approaches are population-based metaheuristics. Unlike a local search method which works on a current single solution, an evolutionary method builds, maintains and evolves a set of solutions. Among the evolutionary approaches proposed for the satisfiability problems, we mention genetic algorithms [12, 22], memetic algorithms [2], ASAP [9], FlipGA [16] and GASAT [15] which appears, currently, as the best evolutionary algorithm for MAX-SAT.

Both evolutionary and SLS methods are incomplete algorithms. They are helpful for finding models of very large instances. For a history and more details on methods for SAT and MAX-SAT problems, the reader can refer to [14].

In this paper, we are interested in scatter search for MAX-SAT. A background on the scatter search framework is given as follows.

The scatter search [6, 7] is a population-based meta-heuristic like genetic algorithms [8]. It is an evolutionary method which constructs solutions by combining others. The approach starts with an initial population (a collection of solutions) generated by using both diversification and improvement strategies, then, a set representing the best solutions (a reference set which includes both diverse and high quality solutions) is selected from the population. This collection of solutions forms the basis for creating new solutions by combining subsets of elements of the current reference solutions. Four methods are involved to achieve the scatter search template:

1. A diversification generation and reference set method that generates diverse solutions, then, improves them and selects the most elite and diverse to create the reference set of solutions.
2. An improvement method which is used in (1) and in (4) to enhance the quality of solutions.
3. A subsets generation method that creates new solutions based on deterministic of subsets of the reference set solutions.
4. A solution combination method that produces solutions inside and outside of the regions spanned by the reference set.

In this paper, we propose a scatter search variant for the MAX − SAT problem. Its algorithmic backbone is a scatter search enhanced with a specific combination operator combined with a SLS improvement strategy. The combination operator is used to identify promising search regions while the SLS performs an intensified search of solutions around these regions. The objective is to achieve a good compromise between intensification and diversification in the search process.

The rest of the paper is organized as follows. The next section gives the scatter search variant for MAX-SAT problems. Experiments on random problems instances and several instances from MAX-SAT libraries are carried out in section 3. Finally, section 4 concludes the work.

2 Scatter Search Variant for MAX-SAT

The scatter search that we propose here starts from good scattered initial solutions created using a diversification method making a population *P*. Then a collection *B* of solutions of size (sizeB) (called reference set) is selected from the current population to participate in the reproduction phase to make other new solutions. The reference set *B* contains, first, B_1 *best solutions* selected according to their fitness value, i.e, solutions having high quality where the quality of a solution is measured by using the objective function which maximizes the number of satisfied clauses. Then, B_2 other solutions from *(P-B)* are added to the reference set *B* to complete it. The B_2 solutions which are the most distant from the best solutions of *B* are called *diverse solutions*. The diversity of a solution is measured by using the Hamming distance which is the number of different bits between two solutions. This selection strategy maintains at each generation a diversified population.

After having selected a set of good and diverse solutions, the specific combination operator is applied on the solutions of the reference set to build new solutions. In order to locate solutions more effectively, a stochastic local search is used as an improvement step.

The new solutions are inserted into the current reference set according to their fitnesses and their diversities values. The new solution is added to the B_1 best solutions and the worst one is removed from it when the new solution improves the quality of the current reference set. Otherwise, if the new solution improves the diversity of the current collection, in this situation, the solution in the reference set having a small diversity is replaced by the new one.

The scatter search process which consists in creating a diversified initial population, in selecting both diverse and best solutions, in applying a specific combination method and a SLS improvement step is stopped when a termination condition is met (a finite number of iterations or a certain amount of time). The main background of our proposed variant is given in the following:

2.1 Solution Representation

A solution is represented by a binary chain X (a n Vector X), whose each component X_i receives the value 0 (False) or 1 (True). It represents an assignment of truth values to the n variables.

2.2 Objective Function

A quality of a solution is measured by using an objective function that computes the number of satisfied clauses.

2.3 Diversity Function

The diversity of a solution is measured by using the Hamming distance which is the number of different bits between two solutions.

Example 1. The Hamming distance between X= 0100 and Y= 0011 is equal to 3.
Example 2. The Hamming distance between X= 0100 and the set of the two solutions (Y = 0011 and Z=0101) is equal to 1 (it equals the minimum between the distance between X and Y and the distance between X and Z).

2.4 Diversification Generator

To generate diverse solutions, the generator creates, from a seed solution X *(an initial solution generated randomly)*, a collection of solutions associated with an integer h, $1 \leq h < n$. Two types of solutions Y and Z are created from the initial solution X using the following formulas:

Type1 solutions:

$$Y_1 = 1 - X_1 \quad (1)$$

$$Y_{1+k\times h} = 1 - X_{1+k\times h}, \quad k = 1, 2..k',\ k' \leq n/h\ ,\quad 1 \leq h < n \quad (2)$$

Type2 solutions: Z is the complement of Y

$$Z_i = 1 - Y_i\ , \quad i = 1, 2..n \quad (3)$$

2.5 Reference Set Selection

The reference set selection strategy focuses on both diverse and quality criteria to choose solutions. The candidate of solutions is selected as follows.

- First, we select from the current population *(P)* a subset of (B_1) highest-quality solutions.
- Second, for each solution *(V)* from the rest of the population *(P-B)*, calculate the Hamming distances between the solution *(V)* and all the solutions in the current *(B)*. We take the minimum value as a diversity value of the solution *(V)*.

– Then, we select from *(P- B)* a number of (B_2) diverse solutions those having a big diversity to complete the reference set *(B)*.

2.6 SLS Intensified Phase

In order to scan the most important part of the whole search space for attaining good solutions, we propose a search technique based on the well known Walksat [24] concepts and augmented by combining diversification, determinism and also randomness which sometimes is very useful to visit new regions.

The proposed SLS consists in deciding with a fixed probability $wp > 0$ whether to do a random walk step *(step 1)* or an extended GSAT step. The extended GSAT step is to decide with probability *(dp-wp)/(1-wp)* whether to do a diversified step in which a randomly selected variable from a randomly selected unsatisfied clause *cl* is flipped (*step 2)* or to do a standard GSAT step *(step 3)*. In a standard GSAT step, a best variable (the one maximizing the number of satisfied clauses when flipped) is picked to be flipped. The SLS process is sketched in Procedure1.

Procedure1. The SLS Intensified search
Input: MAX-SAT formula, a solution *V*, maxflips, *dp, wp*
Output: an improved solution *V*
Begin
  For flip=1 to maxflips **do**
    r= random number between 0 and 1
    If (r < *wp*) **Then** flipvar= pick a random variable /**(step 1)*/
      Else If (r< *dp*) **Then** /**(step 2)*/
        cl= pick a random unsatisfied clause
        flipvar =pick a random variable from *cl*
        Else *flipvar*= pick a best variable /**(step 3)*/
      EndIF
    EndIF
    V = *V* with *flipvar* flipped.
  EndFor
End;

2.7 Combination Method

Here we propose a context-dependent operator which is a specific operator to MAX-SAT problem. It tries to minimize the number of unsatisfied clauses. It takes two solutions (called parents) and produces a new solution (called a child).

To construct the child, the operator, first, examine all the unsatisfied clauses (falsified simultaneously by the two parents) and chooses from the current falsified clause a best variable to fix in the child to verify the clause (i.e. renders the clause *true*).

After having examined all the unsatisfied clauses, we obtain a partial assignment which may be completed using a uniform crossover operator which decides which

parent will contribute for each of the variable values in the child according to a uniform distribution as shown on Procedure2.

Procedure2. The Specific Combination Operator
Input: two parents *Parent1* and *Parent2*
Output: A child, *Child*
Begin
  For All (unsatisfied clauses falsified by parent1 and parent2) **do**
    - let consider, *cl* a falsified clause,
    - select from *cl* the best variable to fix it in the child
  EndFor
  For All (unfixed variables, *var*, in *Child*) **do**
    r= random number between 0 and 1
    If (r < 0.5) **Then** $Child_{var} = Parent1_{var}$
    Else $Child_{var} = Parent2_{var}$
    EndIf
  EndFor
End;

2.8 Scatter Search Algorithm Outline for MAX-SAT

The overall scatter search algorithm is given in Procedure 3.

Procedure3. The scatter search process
Input: an instance of MAX-SAT
Output: an assignment of variables which maximizes the number of satisfied Clauses
Begin
  Generate a diverse population *P* using the diversification method
  do
    Make the Reference Set selection *B*;

    do
      For each two solutions in *B*, use the specific combination method to produce new solutions;
      Apply the SLS to enhance the new trial solutions;
      If (the resulting solution improves the quality) **Then**
        Add it to B_1 and remove the worst solution from *B*.
      Else If (the resulting solution improves the diversity)
        Then Add it to the B_2 and removes from *B* the less diversified one;
      EndIF
      EndIf
      While (all the pair combinations of solutions are examined)
  Until (a certain amount of time is consumed)
  Return the best solution found.
End;

3 Experimental Results

This section lists results of our scatter search variant on some random MAX-2-SAT and MAX-3-SAT instances generated by Borchers et al[2]. The instances are used for a series of paper investigating the performance of MAX-SAT solvers. In addition to these instances, we have taken some instances from the SATLIB[3] and some instances from the Lardeux's home page[4].

First, we compared our scatter search variant (SSV) with a classical genetic algorithm (GA) and a SLS which we have implemented. Then, we compared our methods with Walksat local search method. We also use GASAT and FlipGA which are hybrid genetic algorithms for MAX-SAT in our comparison.

Our source codes are written in C and run on a Pentium- IV 2.8 GHZ, 1GB of RAM under a Linux operating system.

3.1 Comparison with a Classical Genetic Algorithm and a SLS

Tables 1 and 2 compare SSV, SLS[5] and GA[6] on some MAX-SAT problems.

For a fair comparison, the same amount of time (300 second) has been given to the three methods (GA, SLS and SSV).

The parameters of the different algorithms are fixed by empirical studies (not reported here). The fixed values are those for which a good compromise between the quality of the solution obtained by the algorithm and the running time of the algorithm is found.

The SSV parameters are: the population size=100 and the reference set size (B_1=5, B_2=5). The local search improvement phase in the SSV algorithm is given a 60s of time at each call, *wp*=0.3 and *dp*=0.6.

The parameters of the classical genetic algorithm are fixed as: the population size =100, the crossover (uni-point) rate= 0.6 and the mutation rate= 0.1.

The SLS parameters are: *wp*=0.3 and *dp*=0.6.

The whole computation time for each method (GA, SSV and SLS) is the same and equals to 300 second in maximum.

For each method we give in Tables 1 and 2 the best number of violated (*false*) clauses in the best solution found (noted *sol*) and the computation time in second to reach the best solution (noted *time*).

[2] http://www.nmt.edu/ borchers/maxsat.html

[3] http://www.satlib.org/

[4] http://www.info.univ-angers.fr/pub/lardeux/SAT/benchmarks-\\EN.html

[5] SLS is the stochastic local search of procedure 1 executed alone.

[6] GA is a standard genetic algorithm. It uses a standard selection strategy, a one-point crossover operator and a mutation operator and without a local search.

Table 1. Experimental results for some MAX-SAT problems by SSV, GA and SLS. The whole computation time =300s.

Instances.cnf (vars/cls) [7]	SAT[8]	SSV		GA		SLS	
		sol	*time*	*sol*	*time*	*sol*	*time*
par16-5.(1015/3358)	Y	**1**	17.87	485	171.81	3	169.2
par32-1-c (1315/5254)	Y	**6**	185.04	615	142.7	9	273.54
par32-4-c (1333/5326)	Y	**7**	7.90	624	24.21	11	72.32
par32-5-c (1339/5350)	Y	**6**	142.92	614	43.25	9	255.48
par32-5 (3176/10325)	Y	**8**	54.03	1575	157.5	17	295.5
ais10 (181/3151)	Y	**0**	7.51	299	244.37	0	286.53
ais12 (265/5666)	Y	**0**	97.26	642	280.18	1	2.03
Mat25.shuffled (588/1968)	N	**3**	181.81	270	202.15	5	119.15
Mat26.shuffled (744/2464)	N	**3**	251.62	335	228.18	10	127.93
color-10-3 (300/6475)	Y	**0**	61,84	167	142.11	0	85.46
color-18-4 (1296/95904)	Y	**15**	23.75	273	22.26	31	168.82
difp_19_0_arr_rcr (1201/6563)	Y	**14**	20.56	628	98.54	18	240.06
difp_19_1_arr_rcr (1201/6563)	Y	**8**	109.18	631	87.93	20	250.31
difp_19_3_arr_rcr (1201/6563)	Y	**9**	200.37	637	240.07	19	284.00
difp_19_99_arr_rcr (1201/6563)	Y	**10**	139.17	619	116.15	15	249.59
glassy-v399 (399/1862)	Y	**5**	15.78	175	139.56	5	229.01
glassy-v450-(450/2100)	Y	**8**	163.39	197	197.2	10	160.25
hgen2-v500-a (500/1750)	Y	**1**	181.45	171	90.75	3	140.92
hgen2-v500-b (500/1750)	Y	**3**	17.76	172	232.18	4	49.56
f1000 (1000/4250)	Y	**3**	166.56	450	255.10	7	239.82
f2000 (2000/8500)	Y	**5**	215.01	958	146.34	24	231.42
f600 (600/ 2550)	Y	**0**	12.20	236	208.26	3	254.21
jnh201(100/800)	Y	**0**	<0.0001	19	79.67	0	0.01
jnh202(100/800)	N	**1**	0.2660	24	182.09	1	2.78
jnh203(100/800)	N	**1**	0.2340	28	129.43	1	7.73
jnh204.cnf(100/800)	Y	**0**	0.2190	21	165.56	0	0.31
jnh205.cnf(100/800)	Y	**0**	0.0470	25	276.09	0	0.81
jnh206.cnf(100/800)	N	**1**	<0.0001	24	48.12	1	0.68
jnh207.cnf(100/800)	Y	**0**	0.4060	24	106.26	0	1.82
jnh208.cnf(100/800)	N	**1**	0.5780	27	93.57	1	5.25
jnh209.cnf(100/800)	Y	**0**	0.0790	24	99.43	0	2.23

[7] vars (respectively cls) is the number of variables (respectively the number of clauses) of the MAX-SAT instance.

[8] Y (respectively N) meant that the instance is satisfiable (respectively unsatisfiable).

Table2. Experimental results for MAX-2-SAT and MAX-3-SAT problems by SSV, GA and SLS. The whole computation time =300s.

Instances.cnf (vars/cls)	SAT	SSV		GA		SLS	
		sol	*time*	*sol*	*time*	*sol*	*time*
p100 (50/100)	N	**4**	<0.001	8	51.64	4	<0.001
p150 (50/150)	N	**8**	<0.001	15	4.15	8	<0.001
p200 (50/200)	N	**16**	<0.001	23	189.45	16	0.01
p2200 100/200)	N	**5**	<0.001	22	68.29	5	0.04
p2300 (150/300)	N	**4**	0.0160	40	297.37	4	0.016
p2300 (100/300)	N	**15**	<0.001	41	52.84	15	0.047
p2400 (100/400)	N	**29**	0.0320	61	278.71	29	0.047
p2450 (150/450)	N	**22**	0.0160	69	209.93	22	0.063
p250 (50/ 250)	N	**22**	<0.001	31	234.87	22	0.031
p2500(100/500)	N	**44**	0.0160	82	226.70	44	0.047
p2600(150/ 600)	N	**38**	0.01	104	53.98	38	6.906
p2600(100/600)	N	**65**	0.03	105	170.35	65	0.047
p300 (5/300)	N	**32**	<0.001	46	196.48	32	<0.001
p3250 (50/250)	N	**2**	<0.001	11	23.31	2	1.376
p3300 (50/ 300)	N	**4**	0.01	13	127.17	4	0.625
p3350 (50/ 350)	N	**8**	<0.001	19	257.89	8	0.125
p3400 (50/400)	N	**11**	0.01	24	51.34	11	0.141
p3450 (50/450)	N	**15**	0.01	29	87.32	15	0.188
p3750 (150/ 750)	N	**5**	0.03	52	288.23	5	46.70
p400 (50/400)	N	**45**	<0.001	57	220.95	45	0.016
p450 (50/450)	N	**63**	<0.001	77	274.31	63	0.016
p500 (50/500)	N	**66**	<0.001	79	36.37	66	<0.001

From Tables 1 and 2 we see that the SSV consistently outperforms the GA and the SLS providing better results for almost all the problems within shorter times and with significantly smaller deviation from the optimal solutions (it finds the optimal solution in some cases). Our SSV succeeds in finding the optimum solution for some instances. They find the optimum for some problem classes (MAX-2-SAT, MAX-3-SAT, jnh class, ais class, f600, color10-3). For the other classes, they find near good solutions in reasonable time.

It can also be seen that the traditional genetic algorithm (GA) usually fails to find a good solution to MAX-SAT problems for all the classes. It produces poor results. The SSV and the SLS always outperformed the GA and found very encouraging results in a less running time. The genetic algorithm (GA) failure is due to its inherent premature convergence. The performance of our approach is due to the combination of diversification, determinism and also randomness strategies which are very useful to visit new regions which lead to good results.

3.2 Comparison with WalkSat

Walksat is a famous greedy local search for a satisfying assignment of a set of propositional clauses. Figure 1 compares our SSV and Walksat. The results concerning Walksat are taken from [26] where Walksat version 39 is used with a total of 10 million flips. The results are for 20 runs for each instance. The comparison is focused only on the quality of solutions. From Figure 1, we can observe that the SSV

outperforms Walksat on Parity problems. This is due to the combination of diversification and intensification startegies.

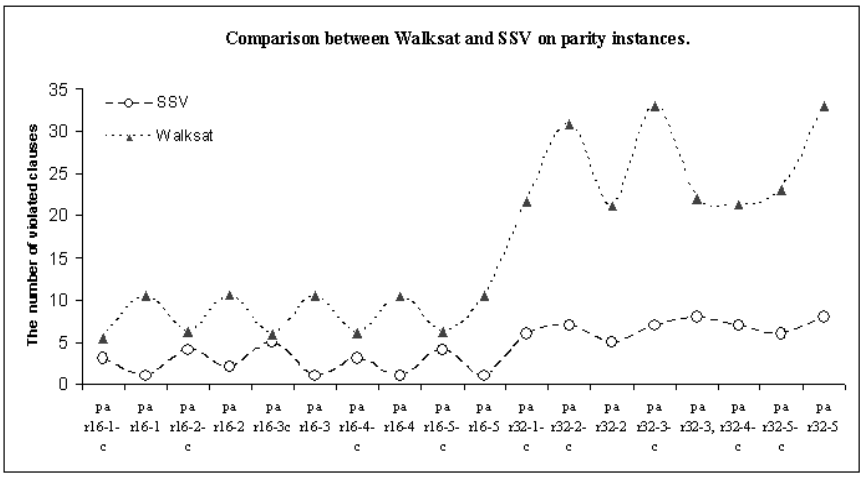


Figure1. Comparison between SSV and Walksat.

3.3 Comparison with GASAT and FlipGA

To fully evaluate the performance of our evolutionary approach (SSV), we have compared it with GASAT and FlipGA. GASAT is a hybrid genetic algorithm proposed recently by [15]. It uses a Tabu search as a local improvement technique and a specific crossover as a combination method. FlipGA is a local genetic algorithm [16] which has been used in [15] for comparison goals.

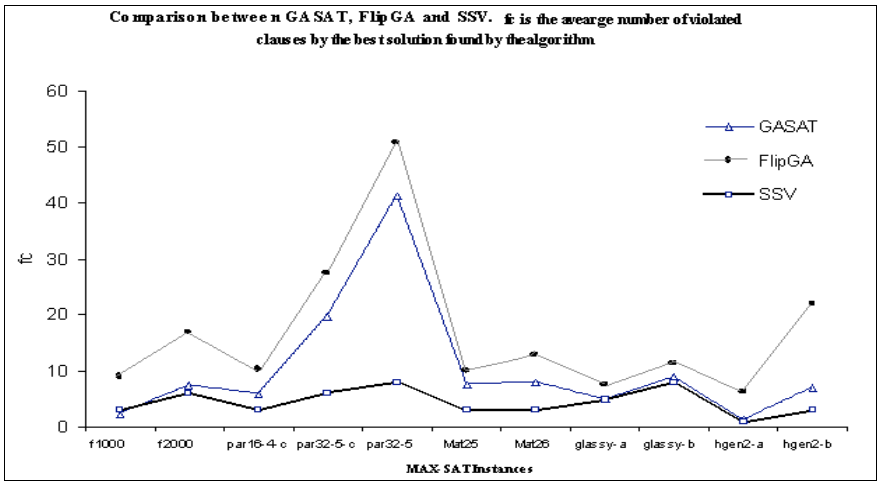


Figure2. Comparison between GASAT, FlipGA and SSV.

Figure 2 compares our SSV, GASAT and FlipGA where The FlipGA parameters are: the population size is 10 individuals, the maximum number of crossover is 10^2 and the mutation rate = 0.9. The GASAT parameters are: the population size is 100 individuals, the sub-population size is 15 individuals, the maximum number of crossover is 10^3 and the tabu search as a local improvement technique with 10^4 flips for each call.

Figure 2 shows that SSV outperforms FlipGA on all instances and GASAT in some cases. SSV produces similar results to those of GASAT on lran instance (f*), glas* and hgen* classes. However, SSV outperforms GASAT on Pariy* and MAT* classes. This confirms the effectiveness of our approach in reaching good quality solutions for various instances compared to the GASAT and FlipGA for MAX-SAT problems.

4 Conclusion

In this paper, we have presented a SLS procedure which consists of three steps: diversification, determinism and randomness. Then, we have proposed a scatter search variant which uses the proposed SLS as an improvement phase, and a specific combination method to produce new solutions. Several experiments are performed on random problems instances and on various instances from MAX-SAT libraries. Comparisons with a classical genetic algorithm, a SLS and with some well-known MAX-SAT solvers show the effectiveness of our new approach to solve MAX-SAT instances. As a perspective work, we will focus on the combination of a stochastic local search and a DPLL- based exact method. Our purpose is to find a good compromise when combining complete approaches with non complete ones.

References

1. T. Alsinet, F. Manya, and J. Planes. ”Improved branch and bound algorithms for Max-SAT*”. In Proceedings of SAT2003*, 2003.
2. D. Boughaci, H Drias and B. Benhamou, Solving Max-SAT Problems Using a memetic evolutionary Meta-heuristic, *In Proceedings of 2004 IEEE CIS 2004*, pp 480-484, 2004..
3. B. Borchers and J. Furman. ”A *two-phase exact algorithm for Max-SAT and weighted Max-SAT problems*”. J. Combinatorial Optimization, 2 (4) 299 -306, 1999.
4. M. Davis, G. Logemann, and D. Loveland. ”*A machine program for theorem proving*”. CACM, 5:394-397, 1962.
5. M.R. Garey et D.S. Johnson. ”Computers *and Intractability, A Guide to the Theory of NP-Completeness*”.W.H. Freeman- Company, San Francisco, 1979.
6. F. Glover. “*Tabu search for nonlinear and parametric optimization (with links to genetic algorithms*)”. Discrete Appl. Math.(49) 231-255. 1994.
7. F. Glover. “*Scatter search and path relinking”.* D. Corne, M.Dorigo, F. Glover, eds. New Ideas in Optimization. McGraw-Hill, New York, 297-316.1999.
8. D. E Goldberg,.”*Genetic Algorithms in search, Optimization Machine Learning*”. Wokingham, Addison-Wesley. 1989.

9. J.Gottlieb, E.Marchiori, and C.Rossi. "*Evolutionary algorithms for the satisfiability problem".* Evolutionary Computation, 10(1) :35-50, 2002.
10. S.D. Givry, J. Larrosa, P. Meseguer, and T. Schiex. "Solving Max-SAT as weighted CSP". *In Proceedings of 9th International Conference on Principles and Practice of Constraint Programming (CP2003),* pps 363-376, 2003.
11. P.Hansen and B.Jaumard. "*Algorithms for the Maximum Satisfiability".* Journal of computing (44) 279-303, 1990.
12. Jin-Kao Hao, F.Lardeux and F.Saubion, "Evolutionary Computing for the Satisability Problem". *Lecture Notes in Computer Science (EvoCOP03),* pp.258-267, UK, Springer, 2003.
13. Holger H. Hoos. "An Adaptive Noise Mechanism for WalkSAT". *AAAI/IAAI 2002*: 655-660,2002.
14. Holger H. Hoos and Thomas Stutzle, "*Stochastic Local Search*", Cambridge, Messachusett: Morgan Kaufman, 2005.
15. F.Lardeux, F. Saubion and Jin-Kao Hao. "*GASAT : A genetic local search algorithm for the satisfiability problem".* Journal of Evolutionary Computation, Summer 2006, vol 14, N°2: 223-253, MIT Press, 2006.
16. E.Marchiori and C.Rossi. "A flipping genetic algorithm for hard 3-SAT problems, *In Proceedings of the Genetic and Evolutionary Computation Conference,* volume 1,pages 393-400, 1999.
17. Chu Min Li, Felip Manya, and Jordi Planes. " Detecting Disjoint Inconsistent Subformulas for Computing Lower Bounds for Max-SAT". *In Proceedings of AAAI 2006*.
18. Chu Min Li, Wen Qi Huang. "Diversification and Determinism in Local Search for Satisfiability". *In Proceedings of SAT 2005*: 158-172, 2005.
19. B.Mazure, L.Sais and E.Greroire.. "A Tabu search for SAT". I*n proceedings the AAAI-97*, pp 281-285, 1997.
20. D. McAllester, B. Selman, and H. Kautz. "Evidence for invariants in local search". *In Proceedings of AAAI-97*, pages 321-326, 1997.
21. P. Mills and E.P.K Tsang. 2000. "*Guided local search for solving SAT and weighted MAX-SAT problems"*, Journal of Automated Reasoning, Special Issue on Satisfiability Problems, Kluwer, Vol.24, 205-223.
22. S. Rana and D. Whitley. "Genetic algorithm behavior in the MAX-SAT domain". *In A. E.Eiben, T. Back, M. Schoenauer, and H.-P. Schwefel, editors, Proceedings of PPSN-V, Fifth International Conference on Parallel Problem Solving from Nature, volume 1498 of Lecture Notes in Computer Science,* pp 785-794. Springer Verlag, Berlin, Germany, 1998.
23. B. Selman, H. Levesque, and D. Mitchell. "A new method for solving hard satisfiability problems". *In Proceedings of the 10th National Conference on Artificial Intelligence*, pp 440- 446. AAAI Press / the MIT Press, Menlo Park, CA, USA, 1992.
24. B. Selman, H. Kautz, and B. Cohen. "Noise strategies for local search". *In Proceedings of AAAI-94*, pp 337-343, 1994.
25. Z.Xing, W.Zhang, MaxSolver: "*An Efficient Exact Algorithm for (Weighted) Maximum Satisfiability"*, Artificial Intelligence ,(2), pages 47-80, 2005.
26. W.Zhang, Rangan and M. Looks. "Backbone guided local search for maximum satisfiability", I*n Proceedings of IJCAI-03*, pp 1179-1184, 2003.

Caractérisation des graphes $b\gamma-$parfaits

Mostafa Blidia
Département de Mathématiques, Faculté des Sciences, Université de Blida.
B. P. 270, Blida. E-mail: mblidia@hotmail.com

Noureddine Ikhlef-Eschouf
Département de Mathématiques, Faculté des Sciences, Université de Blida.
B. P. 270, Blida. E-mail: nour_echouf@yahoo.fr

Frédéric Maffray
C.N.R.S., Laboratoire Leibniz-IMAG, 46 Avenue Félix Viallet,
38031 Grenoble Cedex, France. e-mail: frederic.maffray@imag.fr

Résumé

Une coloration dominante est une coloration propre telle que toute classe de couleur i contient un sommet adjacent à au moins un sommet de chaque classe de couleur $j \neq i$. Le nombre $b-$chromatique, noté $b(G)$, est le nombre maximum de classes de couleurs dans une coloration dominante. Une $k-$coloration de Grundy d'un graphe G est une coloration séquentielle propre utilisant k couleurs telle que tout sommet reçoit la première couleur disponible qui n'existe pas dans son voisinage. Le nombre de *Grundy* d'un graphe G, noté $\gamma(G)$, est le nombre maximum de couleurs nécessaires pour colorer les sommets de G avec une coloration de *Grundy*. Un graphe G est dit $b\gamma-$parfait si $b(H) = \gamma(H)$ pour tout sous graphe induit H de G. Dans ce papier, on présente une nouvelle borne inférieure pour le nombre $b-$chromatique dans la classe des graphes sans P_4 et montre que ce paramètre peut être calculé en temps polynamial dans cette classe de graphes ensuite, on caractérise les graphes $b\gamma-$parfaits.

Mots clés: Coloration, $b-$Coloration, coloration de *Grundy*

Ces travaux communs ont été réalisés grâce à l'accord programme CMEP-Tassili numéro: 05 MDU 639.

1 Introduction

Soit $G = (V, E)$ un graphe simple avec V ensemble des sommets et E ensemble des arêtes, pour $A \subset V$, on désigne par $G[A]$ le sous graphe de G engendré par A. On désigne par $G = (X, Y, E)$ le graphe biparti connexe où X, Y sont les classes de la bipartition de G. Pour un sommet v de G, le voisinage ouvert est défini par l'ensemble $N_G(v) = \{u \in V(G) : (u, v) \in E\}$ et le voisinage fermé est défini par $N_G[v] = N_G(v) \cup \{v\}$. Etant donné un graphe H, un graphe G est dit sans H si G ne contient pas H comme sous-graphe induit. Pour plus de détails sur la terminologie utilisée, on renvoit le lecteur à [1].

Une $k-$coloration de Grundy d'un graphe G est une coloration propre utilisant k couleurs vérifiant la propriété suivante : chaque sommet v, coloré par une couleur i , $1 \leq i \leq k$, doit être adjacent à au moins $i - 1$ sommets colorés par chacune des couleurs j telles que $1 \leq j \leq i - 1$. Le nombre de Grundy d'un graphe G, noté $\gamma(G)$, est le nombre maximum de couleurs nécessaires pour une coloration de Grundy de G, pris sur toute les colorations de Grundy de G. Ce paramètre a été introduit par C. Christen et S. Selkow [3]. On désigne par $\omega(G), \chi(G), b(G)$, et $\gamma(G)$ respectivement la taille d'une clique maximum de G, le nombre chromatique de G, le nombre $b-$chromatique de G et le nombre de Grundy de G. Pour deux nombres distincts α et β de $\{\omega, \chi, b, \gamma\}$, un graphe G est dit $\alpha\beta-$parfait si et seulement si $\alpha(H) = \beta(H)$ pour tout sous graphe induit H de G. Par exemple, un graphe G est dit $b\gamma-$parfait si et seulement si $b(H) = \gamma(H)$ pour tout sous graphe induit H de G. La notion de graphes $b\chi-$parfaits a été introduite récemment par Hoàng et Kouider [4] et a fait l'objet d'un article publié par les mêmes auteurs dans lequel ils ont caractérisé quelques classes de graphes $b\chi-$parfaits..

En général, pour un graphe G quelconque, le nombre $b-$*chromatique* et le nombre de *Grundy* sont incomparables. Par exemple, $\gamma(2D) = 3 < b(2D) = 4$ et $\gamma(P_4) = 3 > b(P_4) = 2$, où $2D$ et P_4 désignent respectivement, l'union de deux diamants (voir figure 3) et une chaîne élémentaire d'ordre 4. Il est intéressant donc de comparer ces deux paramètres pour des classes de graphes particulières et de caractériser la famille des graphes $b\gamma-$parfaits.

Pour qu'un graphe G admette une coloration dominante avec k couleurs, il doit avoir au moins k sommets de degré supérieur à $k - 1$, de cette observation, Irving et Manlove [5, 7] ont défini le $m-$degré qui donne une borne supérieure du nombre $b-$chromatique :

Définition 1 *Soit les sommets $v_1, v_2, ...v_n$ d'un graphe G ordonnés tels que:*

$d(v_1) \geq d(v_2) \geq ... \geq d(v_n)$, *où* $d(v_i)$ *est le degré de* v_i. *Le* $m-$*degré de* G, *noté* $m(G)$ *est défini par :* $m(G) = max\{i : d(v_i) \geq i-1\}$.

Définition 2 *On appelle joint de deux graphes* G *et* H, *noté* $G \vee H$, *le graphe obtenu en prenant une copie de* G *et une copie de* H *et en reliant tous les sommets de* G *à tous les sommets de* H.

Définition 3 *Le diamant noté* D *(ou* $K_4 - e$*) est le graphe obtenu après avoir enlevé une arête à la clique d'ordre quatre.*

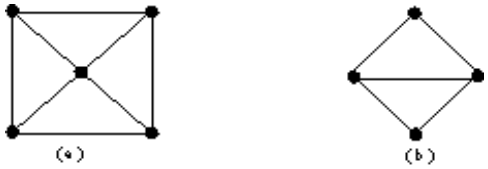


Figure 1. (a) Le joint $K_1 \vee C_4$ (b) Le diamant

Définition 4 *Soient* G *et* H *deux graphes disjoints. Alors* $G + H$ *désigne l'union de* G *et* H. *Pour un entier* k *donné,* kG *désigne l'union de* k *copies de* G.

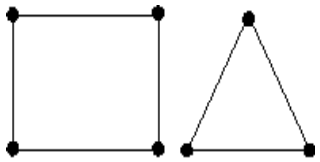


Figure 2. L'union de C_4 et C_3 $(C_4 + C_3)$

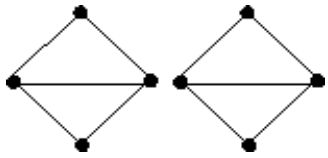


Figure 3. Deux diamants $(2D)$

Dans [4], C.T. Hoàng et M. Kouider ont donné les résultats suivants

Lemme 5 *[4] Soit donné un entier* $k \geq 1$ *et soient* G_1 *et* K_k *deux graphes de sommets disjoints où* K_k *est une clique d'ordre* k, *et soit* $G = G_1 + K_k$. *Alors on a* $b(G) = max(b(G_1), k)$.

Lemme 6 *[4] Soient G_1 et G_2 deux graphes de sommets disjoints. Alors le joint $G_1 \vee G_2$ a $b(G_1 \vee G_2) = b(G_1) + b(G_2)$.*

Dans [6], M. Kouider and M. Mahéo ont donné d'autres résultats relatifs à la $b-$coloration des graphes bipartis complets et graphes non connexes.

Proposition 7 *[6] Pour tout $n, p \in \mathbb{N}$, on a $b(K_{n,p}) = 2$, où $K_{n,p}$ est un graphe biparti complet.*

Proposition 8 *[6] Si G est un graphe non connexe avec les composantes connexes $C_1, C_2, .., C_p$. Alors on a $b(G) \geq max(b(C_i), 1 \leq i \leq p)$.*

2 Résultats préliminaires

Nous établissons les lemmes suivants qui nous seront utiles pour caractériser les graphes $b\gamma-$parfaits.

Lemme 9 *Soient G_1 et G_2 deux graphes bipartis complets et de sommets disjoints, et soit $G = G_1 + G_2$. Alors on a $b(G) = 2$.*

Preuve. Si G est un graphe biparti complet connexe alors, d'après la proposition 7, $b(G) = 2$. Supposons maintenant que $G = G_1 + G_2$ avec G_1 et G_2 deux graphes bipartis complets. D'après la proposition 8 on a, $b(G) \geq max\{b(G_i), 1 \leq i \leq 2\}$. Si $b(G) > max\{b(G_i), 1 \leq i \leq 2\} = 2$ alors G_1 (ou G_2) contient deux sommets $b-$dominants de couleur distinctes. Sans perte de généralité, on peut supposer que, G_1 contient deux sommets $b-$dominants. Comme $G_1 = (X_1, Y_1, E_1)$ est un graphe biparti complet alors, on distingue deux cas:

Cas 1 : Les deux sommets $b-$dominants appartiennent à la même bipartition X_1. Puisque $b(G) \geq 3$ alors, toutes les couleurs apparaissent dans Y_1. Comme G_1 est biparti complet alors, la coloration obtenue n'est pas propre.

Cas 2 : Les deux sommets $b-$dominants n'appartiennent pas à la même bipartition. Comme $b(G) \geq 3$ et G_1 est un graphe biparti complet alors, la coloration obtenue n'est pas propre. ■

Remarque 10 *Si $G_1, G_2, ..., G_p$, $p \geq 3$ sont des graphes bipartis complets. Alors on a $b(G_1 + G_2 + ...G_p) \geq 2$. Par exemple, $b(3P_3) = 3$.*

Lemme 11 *Soient G_1 et G_2 deux graphes de sommets disjoints. Alors le joint $G_1 \vee G_2$ a $\gamma(G_1 \vee G_2) = \gamma(G_1) + \gamma(G_2)$.*

Preuve. Il est clair que, dans toute coloration de $G_1 \vee G_2$, aucune couleur ne peut apparaître dans G_1 et G_2 à la fois. Donc $\gamma(G_1 \vee G_2) \geq \gamma(G_1) + \gamma(G_2)$. Si dans une coloration de Grundy de $G_1 \vee G_2$, $\gamma(G_1 \vee G_2) > \gamma(G_1) + \gamma(G_2)$ alors, on aura plus de $\gamma(G_1)$-couleurs (ou $\gamma(G_2)$-couleurs) qui se présentent dans une coloration de Grundy de G_1(ou G_2), contradiction avec la définition de $\gamma(G_1)$ (ou $\gamma(G_2)$). ■

Lemme 12 *Soient G_1 et G_2 deux graphes de sommets disjoints, et soit $G = G_1 + G_2$. Alors, on a:*

(**i**) $\gamma(G) = \gamma(G_1 + G_2) = \max\{\gamma(G_1), \gamma(G_2)\}$.
(**ii**) Si G_2 est une clique d'ordre k, alors on a: $\gamma(G) = \max\{\gamma(G_1), k\}$.

Preuve. **i**) Soient G_1 et G_2 deux graphes de sommets disjoints, et soit $G = G_1 + G_2$. Tout d'abord, il est clair que $\gamma(G) = \gamma(G_1 + G_2) \geq \max\{\gamma(G_1), \gamma(G_2)\}$. Soit donnée une coloration de Grundy de G avec $\gamma(G)$ couleurs et supposons que $\gamma(G) = \gamma(G_1 + G_2) > \max\{\gamma(G_1), \gamma(G_2)\}$. Si $\max\{\gamma(G_1), \gamma(G_2)\} = \gamma(G_1)$ alors, sans perte de généralité, on peut supposer que, les couleurs $1, 2, ..., \gamma(G_1)$ apparaissent dans G_1. Ceci implique que les couleurs $\gamma(G_1) + 1, \gamma(G_1) + 2, .., \gamma(G)$ apparaissent dans G_2 car sinon on aura une contradiction avec la définition de $\gamma(G_1)$. De ce fait, il existe au moins une couleur c, $1 \leq c \leq \gamma(G_1)$ qui ne figure pas dans G_2, sinon $\gamma(G_2) \geq \gamma(G_1) + 1$; contradiction avec le fait que $\gamma(G_1) = \max\{\gamma(G_1), \gamma(G_2)\}$. D'après la définition de la coloration de Grundy, tous les sommets de couleurs $\gamma(G_1) + 1, \gamma(G_1) + 2, .., \gamma(G)$ sont adjacents à un sommet de G_1 de couleur c. Contradiction avec le fait que G est non connexe.

ii) il est facile de vérifier que, si G_2 est une clique d'ordre k, alors on a $\gamma(G) = \max\{\gamma(G_1), k\}$ ■

3 Graphes sans P_4

Nous rappelons d'abord le théorème de D. Seinsche [8]

Théorème 13 *[8] Soit $G = (V, E)$ un graphe. Alors les deux conditions suivantes sont équivalentes:*

(**1**) G est sans P_4.
(**2**) Pour tout ensemble A de V tel que $|A| \geq 2$, ou bien $G[A]$ ou bien son complémentaire est non connexe.

3.1 Borne inférieure du nombre $b-$chromatique

Théorème 14 *Si G est un graphe sans P_4. Alors on a $\gamma(G) \leq b(G)$.*

Preuve. Nous montrons ce théorème par induction sur le nombre de sommets de G. Tout d'abord, il est clair que, la proposition est triviale pour $|V| = 1$. Supposons qu'elle est vraie pour tout ordre k, $k \leq |V| - 1$. Comme G sans P_4 alors, d'après le thorème 13, on distingue deux cas:

Cas 1 : Si $\overline{G}$ est non connexe alors G est le joint de deux graphes G_1 et G_2. D'après les lemmes 6 et 11, $b(G) = b(G_1 \vee G_2) = b(G_1) + b(G_2) \geq \gamma(G_1) + \gamma(G_2) = \gamma(G_1 \vee G_2) = \gamma(G)$.

Cas 2 : Si G est non connexe alors G est l'union de p composantes connexes, $p \geq 2$. D'après la proposition 8 et le lemme 12, $b(G) = b(G_1 + G_2 + .. + G_p) \geq \max\{b(G_i;\ 1 \leq i \leq p\} \geq \max\{\gamma(G_i;\ 1 \leq i \leq p\} = \gamma(G)$. ■

Remarque 15 *La réciproque est fausse. En effet, dans la figure* 4, $b(G) = 5 > \gamma(G) = 4$ *alors que G contient un P_4 induit .*

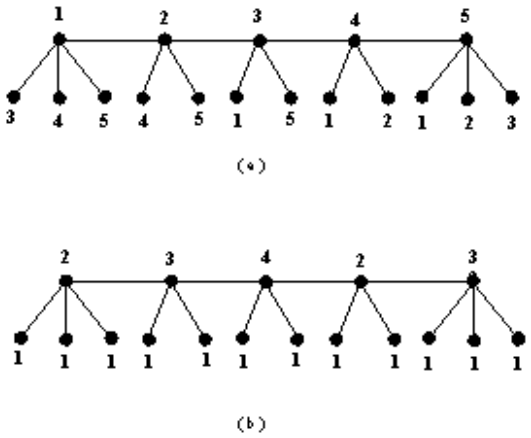


FIGURE 4.(*a*) *Une* (5)*b-coloration .* (*b*) *Une* 4*-coloration de Grundy*

3.2 Calcul du nombre b-chromatique

Nous montrons ici comment calculer le nombre b-chromatique d'un graphe G sans P_4. Pour cela, nous posons quelques définitions supplémentaires. Dans une coloration donnée, nous disons qu'un sommet est un *b-sommet* s'il a un voisin dans chaque classe de couleur autre que la sienne; et nous disons

qu'une classe de couleur est une *b-classe* si elle contient un b-sommet. Pour tout entier q tel que $\chi(G) \leq q \leq |V(G)|$, soit $h(G,q)$ le nombre maximum de b-classes dans les colorations ayant exactement q couleurs. Avec cette notation, on voit que $b(G) = \max\{q \mid h(G,q) = q\}$. Donc pour calculer $b(G)$, il suffit de calculer $h(G,q)$ pour chaque entier $q \in \{\chi(G), \ldots, |V(G)|\}$. Par le théorème de Seinsche [8], trois cas se présentent.

Cas 1 : *G a un seul sommet.* Dans ce cas on a $q = 1$ et $h(G,1) = 1$.

Cas 2 : *G n'est pas connexe.* Donc G est l'union de deux graphes G_1, G_2. Montrons que $h(G,q) = \min\{q, h(G_1,q) + h(G_2,q)\}$.

Il est clair que $\chi(G) = \max\{\chi(G_1), \chi(G_2)\}$. Considérons une coloration c de G qui a q couleurs et $r = h(G,q)$ b-classes. Pour $j = 1,2$, soit r_j le nombre des b-classes de c qui sont incluses dans G_j; donc $r = r_1 + r_2$. Supposons que $r_j > 0$. Donc G_j contient un b-sommet u. Puisque u a des voisins de toutes les couleurs autres que $c(u)$, et puisque tous ses voisins sont dans G_j, le graphe G_j a des sommets de toutes les q couleurs, et tous les b-sommets de c qui sont dans G_j sont des b-sommets de la restriction de c à G_j, d'où $h(G_j,q) \geq r_j$. On en déduit que $h(G,q) = r = r_1 + r_2 \leq h(G_1,q) + h(G_2,q)$.

Réciproquement, pour $j = 1,2$, soit c_j une coloration de G_j avec q couleurs et $r_j = h(G_j,q)$ b-classes. Quitte à renommer les couleurs, on peut supposer que les b-classes de c_1 sont les couleurs $1, \ldots, r_1$ et que les b-classes de c_2 sont $q - r_2 + 1, \ldots, q$. Par conséquent, en combinant c_1 et c_2 nous obtenons une coloration de G avec q couleurs et $\min\{q, r_1 + r_2\}$ b-classes. Donc $h(G,q) \geq \min\{q, h(G_1,q) + h(G_2,q)\}$. Les deux inégalités impliquent l'égalité annoncée.

Cas 3 : *$\overline{G}$ n'est pas connexe.* Donc G est le joint de deux graphes G_1, G_2. Montrons que $h(G,q) = \max\{h(G_1,q_1) + h(G_2,q_2) \mid q_1 \geq \chi(G_1),\ q_2 \geq \chi(G_2),\ q_1 + q_2 = q\}$.

Il est clair que $\chi(G) = \chi(G_1) + \chi(G_2)$. Considérons une coloration c de G qui a q couleurs et $r = h(G,q)$ b-classes. Pour $j = 1,2$, soient q_j et r_j respectivement le nombre de couleurs et de b-classes de c qui se trouvent dans G_j. Puisqu'une couleur ne peut pas être présente à la fois dans G_1 et dans G_2, on a $q = q_1 + q_2$ et $r = r_1 + r_2$. Tout b-sommet de c qui se trouve dans G_j est un b-sommet de la restriction de c à G_j, d'où $r_j \leq h(G_j,q_j)$ $(j = 1,2)$. Donc $h(G,q) = r_1 + r_2 \leq h(G_1,q_1) + h(G_2,q_2)$.

Réciproquement, soient q_1, q_2 deux entiers tels que $q_1 \geq \chi(G_1)$, $q_2 \geq \chi(G_2)$ et $q_1 + q_2 = q$. Pour $j = 1,2$, posons $r_j = h(G_j,q_j)$ et soit c_j une coloration de G_j avec q_j couleurs et r_j b-classes. En attribuant des ensembles

de couleurs disjoints à c_1 et à c_2, leur combinaison est une coloration c de G avec q couleurs. Puisque chaque sommet de G_1 est adjacent à chaque sommet de G_2, il suit que tout sommet de G_1 qui est un b-sommet de c_1 est aussi un b-sommet de c. Donc c possède $r_1 + r_2$ b-classes. Puisque ceci est vrai pour tous les choix possibles de q_1, q_2, nous obtenons $h(G, q) \geq \max\{h(G_1, q_1) + h(G_2, q_2) \mid q_1 \geq \chi(G_1), q_2 \geq \chi(G_2), q_1 + q_2 = q\}$. Les deux inégalités impliquent l'égalité annoncée.

On observe que les opérations ci-dessus prennent un temps polynomial. Dans le cas 2, le calcul de $h(G, q)$ est immédiat à partir de $h(G_1, q)$ et $h(G_2, q)$. Dans le cas 3, on a au plus $|V(G)|$ couples (q_1, q_2) à examiner. De plus, chacun des cas 2 et 3 réduit le calcul de $h(G, q)$ au calcul sur deux sous-graphes disjoints. On a donc au plus $|V(G)|$ occurrences des cas 2 et 3. Au total, le temps de calcul est donc $O(|V(G)|^2)$ pour chaque valeur de q, et donc $O(|V(G)|^3)$ pour toutes les valeurs possibles et pour en déduire la valeur de $b(G)$.

4 Caractérisation des graphes $b\gamma-$parfaits

Lemme 16 *Soient G et H deux graphes sans P_4 et sans $3P_3$ avec $G = G_1 + G_2$ et $H = G_2 + G_3$, où G_1 est un graphe biparti complet, G_2 est un graphe contenant un diamant et G_3 est le joint d'un seul sommet x_0 avec p cliques de sommets disjoints, $p \geq 2$. Alors, on a :*

i) $b(G) = max\{b(G_1), b(G_2)\} = b(G_2)$
ii) $b(H) = max\{b(G_2), b(G_3)\}$.

Preuve. **i**) Tout d'abord, il clair que $b(G_1) = 2$ et $b(G_2) = k \geq 3$. Supposons que $b(G) = l > \max\{b(G_1), b(G_2)\} = k$. Soient $u_1, u_2, ..., u_l$, les sommets $b-$dominants de G de couleurs $1, 2, .., l$. Si G_2 contient l sommets $b-$dominants de couleurs distinctes alors, $b(G) = l > \max\{2, l\} = l$, contradiction. Donc G_2 contient au plus $l - 1$ sommets $b-$dominants de couleurs distinctes; et comme G_1 est un graphe biparti complet et $l > 3$ alors, il ne peut contenir qu'un seul sommet $b-$dominant. Sans perte de généralité, on peut supposer que, le seul sommet $b-$dominant dans G_1 est u_l de couleur l. Ainsi, le graphe G_2 contient exactement $l - 1$ sommets $b-$dominants $u_1, u_2, ..., u_{l-1}$ de couleurs $1, 2, .., l-1$ différentes de la couleur l. Aucun sommet de G_2 de couleur l ne peut être adjacent à tous les sommets $u_1, u_2, ..., u_{l-1}$, sinon il sera $b-$dominant de couleur l, et $b(G_2) = l$, contradiction.

On montre maintenant qu'il existe dans G_2 deux sommets $b-$dominants u_i et u_j de couleur i et j, $i \neq j$ tels que u_i est adjacent à un sommet $v \in V(G_2)$ de couleur l et u_j est adjacent à un autre sommet $w \in V(G_2)$ de couleur l où u_i est non adjacent à w et u_j; et u_j est non adjacent à v.

En effet, il suffit de prendre un sommet v de couleur l qui a le plus de voisins $b-$dominants de couleurs distinctes dans G_2. Comme v ne peut pas être adjacent à tous les sommets $b-$dominants alors, il existe un sommet $b-$dominant u_j de couleur j adjacent à un autre sommet w de couleur l et non adjacent à v. Dans ce cas, il existe un sommet $u_i \in N(v)$ tel que u_i est non adjacent à w, un tel sommet existe sinon on a une contradiction avec le choix de v. Comme le graphe G est sans P_4 alors, u_i n'est pas adjacent à u_j, sinon $[vu_iu_jw]$ serait un P_4. D'autre part, le sommet v ne peut pas être adjacent à tous les voisins de u_i, sinon v serait un sommet $b-$dominant de couleur l. Donc il existe un sommet x voisin de u_i et non voisin de v. De même pour le sommet u_j, il existe un sommet y voisin de u_j et non voisin de w. Si $x = y$ alors $[vu_ixu_jw]$ serait un P_5, contradiction. Donc $x \neq y$; le sommet x ne peut pas être adjacent aux sommets u_j, w et y sinon $[u_jxu_iv]$, $[wxu_iv]$ et $[xyu_jw]$ induisent dans chacun des cas un P_4. De même, y n'est pas relié aux sommets u_j et v. Donc G_2 contient un $2P_3$, $[xu_iv]$ et $[wu_jy]$; d'où le graphe G contient un $3P_3$ induit, contradiction. Donc on a bien $b(G) = max\{b(G_1), b(G_2)\}$.

ii) Il est facile de constater que, le $m-$degré de G_3 est égal à $\omega(G_3)$, donc $b(G_3) = \omega(G_3)$. Si $b(H) = l > max\{b(G_2), b(G_3)\} = max\{b(G_2), \omega(G_3)\} = k$ alors $\omega(G_3) \leq l-1 = b(H)-1$. Dans ce cas, on peut vérifier que, pour tout sommet z de G_3, $z \neq x_0$, $d(z) \leq \omega(G_3) - 1 \leq b(H) - 2$. En conséquence, aucun sommet de G_3 autre que x_0 ne peut être un sommet $b-$dominant. Ceci implique que G_2 contient au moins $l - 1$ sommets $b-$dominants. En utilisant un raisonnement similaire à (i), on peut montrer que, $b(H) = max\{b(G_2), b(G_3)\}$. ■

Définition 17 *Un graphe G est dit $b\gamma-$imparfait minimal, si tout sous graphe propre $H \subset G$ est $b\gamma-$parfait et le graphe G n'est pas $b\gamma-$parfait.*

Théorème 18 *Soit $G = (V, E)$ un graphe, alors les deux conditions suivantes sont equivalentes*

i) G est $b\gamma-$parfait
ii) G est sans P_4, sans $3P_3$ et sans $2D$.

Preuve. $(i) \Rightarrow (ii)$, $b(P_4) = 2 < \gamma(P_4) = 3$, $b(3P_3) = 3 > \gamma(3P_3) = 2$ et $b(2D) = 4 > \gamma(2D) = 3$. Si G est $b\gamma-$parfait alors, il ne doit pas contenir les graphes isomorphes à P_4, $3P_3$ et $2D$ comme sous graphes induits.

$(ii) \Rightarrow (i)$, soit G un graphe sans P_4, sans $3P_3$ et sans $2D$ et supposons que G est $b\gamma-$imparfait minimal. Comme G est sans P_4 alors d'après le théorème de Seinsche [8], on distingue deux cas:

Cas 1 : Si $\overline{G}$ est non connexe alors G est le joint de deux graphes G_1 et G_2. D'après les lemmes 6 et 11, on a: $b(G) = b(G_1 \vee G_2) = b(G_1) + b(G_2)$ et $\gamma(G) = \gamma(G_1 \vee G_2) = \gamma(G_1) + \gamma(G_2)$. Puisque G_1 et G_2 sont $b\gamma-$parfaits alors $b(G) = \gamma(G)$; contradiction avec le fait que G est $b\gamma-$imparfait minimal.

Cas 2 : Si G n'est pas connexe alors, il contient k composantes connexes $G_1, G_2, ..., G_k$, $k \geq 2$. Deux sous cas sont à distinguer .

Cas 2.1 : $k \geq 3$. Comme G est sans $3P_3$ alors, il existe une composante connexe sans P_3 (G_3 par exemple). Ceci implique que G_3 est une clique. D'après les lemmes 5 et 12, $b(G) = max\{b(G_3), b(G - G_3)\}$ et $\gamma(G) = max\{\gamma(G_3), \gamma(G - G_3)\}$. Comme G_3 et $G - G_3$ sont $b\gamma-$parfaits alors $b(G) = \gamma(G)$; contradiction.

Cas 2.2 : $k = 2$. Puisque G est sans $2D$ alors, on peut supposer que G_1 est sans diamant, et on peut supposer aussi que G_1 n'est pas une clique (sinon voir le cas où $k \geq 3$). En conséquence, le graphe G_1 contient une chaîne P_3. Comme G_1 est sans P_4 alors, il est le joint de deux graphes $G[A]$ et $G[B]$. Les sous graphes $G[A]$ et $G[B]$ sont sans P_3 induit car sinon G_1 contient un diamant. Par conséquent, la chaîne existante dans G_1 est de la forme suivante:

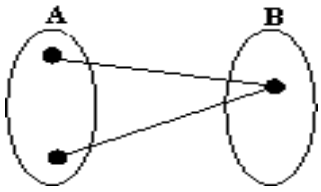


Figure 4. Une chaîne d'ordre 3

L'ensemble B est un stable car sinon G_1 contient un diamant. On distingue donc deux sous sous cas:

Cas 2.2.1 : Si $|B| \geq 2$ alors A est un stable sinon G_1 contient un diamant. Ainsi, G_1 est un graphe biparti complet. D'après la proposition 7 et le lemme 12, on a $b(G_1) = 2$ et $\gamma(G_1) = 2$.

– Si G_2 est sans diamant alors, on peut montrer de la même façon qu'il est biparti complet. Donc $b(G_2) = \gamma(G_2) = 2$. D'après les lemmes 9

et 12, $b(G) = b(G_1 + G_2) = max\{b(G_1), b(G_2)\} = 2$, et $\gamma(G) = \gamma(G_1 + G_2) = \max\{\gamma(G_1), \gamma(G_2)\} = 2$. Comme G_1 et G_2 sont $b\gamma-$parfaits alors $b(G) = \gamma(G)$; contradiction.

– Si G_2 contient un diamant alors, en utilisant le lemme 16, on trouve $b(G) = \max\{b(G_1), b(G_2)\}$. D'après le lemme 12, on a $\gamma(G) = \max\{\gamma(G_1), \gamma(G_2)\}$. D'où $b(G) = \gamma(G)$; contradition.

Cas 2.2.2 : Si $|B| = 1$ alors l'ensemble A est sans P_3. En conséquence, l'ensemble A est l'union de p cliques de sommets disjoints, $p \geq 2$.

– Si G_2 est sans diamant alors, il est de même structure que G_1. On peut vérifier que $b(G_1) = \omega(G_1)$, $b(G_2) = \omega(G_2)$, et $b(G_3) = \omega(G_3)$ de plus, le $m-$degré de G est égal à $\omega(G)$. De ce fait, $b(G) = \omega(G) = \max\{\omega(G_1), \omega(G_2)\} = \max\{b(G_1), b(G_2)\}$, et d'après le lemme 12, on a $\gamma(G) = \max\{\gamma(G_1), \gamma(G_2)\}$, donc $b(G) = \gamma(G)$; contradiction.

– Si G_2 contient un diamant alors, d'après le lemme 16, on trouve $b(G) = \max\{b(G_1), b(G_2)\}$, et comme $\gamma(G) = \max\{\gamma(G_1), \gamma(G_2)\}$ alors $b(G) = \gamma(G)$; contradiction. ■

Corollaire 19 *Si G est un graphe biparti alors, les deux conditions suivantes sont équivalentes:*

i) G est $b\gamma-$parfait

ii) G est sans P_4et sans $3P_3$

Preuve. Simple à vérifier puisqu'un biparti ne contient pas de diamant. ■

5 Graphes $b\psi-$parfaits

Définition 20 *Une $a-$coloration (ou coloration complète) est une coloration propre telle que pour toute paire de couleurs distinctes i et j, il existe deux sommets adjacents de couleur i et j. Le nombre $a-$chromatique, noté $\psi(G)$, est le nombre maximum de classes de couleurs dans une $a-$coloration.*

Dans [3], C.A. Christen et S.M. Selkow ont caractérisé les graphes $\psi\chi$-parfaits.

Théorème 21 *[3]Pour tout graphe G, les assertions suivantes sont équivalentes:*

i) G est $\psi\chi-$parfait

ii) G ne contient aucun sous graphe induit isomorphe à l'un des graphes suivant: $P_4, P_3 + P_2$ et $3P_2$.

En utilisant les théorèmes 18 et 21, on peut déduire le corollaire suivant:

Corollaire 22 *Soit G un graphe, alors les trois assertions suivantes sont équivalentes:*

i) G est $b\psi-$parfait
ii) G est sans P_4, sans $3P_2$ et sans $P_3 + P_2$
iii) G est $\psi\chi-$parfait.

Preuve. $(i) \Rightarrow (ii)$, $b(P_4) = 2 < \psi(P_4) = 3$, $b(3P_2) = 2 < \psi(3P_2) = 3$ et $b(P_3 + P_2) = 2 < \psi(P_3 + P_2) = 3$. Si G est $b\psi-$parfait alors, il ne doit pas contenir les graphes isomorphes à P_4, $3P_2$ et $P_3 + P_2$ comme sous graphes induits.
$(ii) \Rightarrow (iii)$,d'après le théorème 21.
$(iii) \Rightarrow (i)$, trivial, car pour tout graphe G, $\chi(G) \leq b(G) \leq \psi(G)$. ■

References

[1] C. Berge. Graphs, North Holland, 1985.

[2] R. Brooks. On coloring the nodes of a network. Math. Proc. Cambridge Philos. Soc., 37 :194-197, (1941)

[3] C.Christen, S. Selkow, Some Perfect Coloring Properties of Graphs, J. Comb. Theory, Ser. B 27, 49-59 (1979)

[4] C.T. Hoàng and M. Kouider. On the b-dominating coloring of graphs. Discrete Applied Mathematics, 152: 176-186, (2005).

[5] R. W. Irving and D. F. Manlove. The b-chromatic number of graphs. Dicrete Appl. Math. 91: 127-141, (1999).

[6] M. Kouider and M. Mahéo. Some bounds for the b-chromatic number of a graph. Discrete Math., 256: 267-277, (2002).

[7] D. F. Manlove. Minimaximal and maximinimal optimisation problems: a partiel order-based approach. PhD thesis, technical report tr-1998-27 of the Computing Science Department of Glasgow University, 1998.

[8] D. Seinche. On a property of the class of n-colorable graphs, J.Comb. Theory Ser. B 16 (1974), 191-193

Modélisation et Simulation de la Stérilisation Externe pour le Nouvel Hôpital d'ESTAING

Fateh Mebrek[1], Khaled Belkadi [1,2], Michel Gourgand[1], Alain Tanguy[1],

[1] LIMOS, Université Blaise Pascal, Complexe des Cézeaux, 63173 Aubière Cedex, France
{mebrek, belkadi, gourgand, tanguy}@isima.fr

[2] Département d'informatique, USTO-MB, BP. 1505 Oran M'Naouer, 31000, Algérie
belkadi1999@yahoo.com

Résumé. La stérilisation du matériel médical occupe une place importante dans les hôpitaux. Le but de cet article est de modéliser et de simuler la stérilisation externe du matériel médical du nouvel hôpital d'Estaing (NHE) de Clermont Ferrand. En effet l'emploi fréquent de matériel à usage unique, principalement les seringues, aiguilles et cathéters, n'a pas entraîné la suppression de la nécessité de stériliser dans l'hôpital les instruments, le linge opératoire et autres. La stérilisation d'un objet est la destruction ou l'inactivation irréversible de tous les micro-organismes qui se trouvent dans ou sur cet objet. Pour cela, nous avons utilisé en premier lieu, l'outil ARIS pour spécifier le modèle de connaissance et en deuxième lieu la simulation par SIMULA et Witness pour le passage vers le modèle d'action. Nous avons obtenu pour ces deux simulateurs des résultats que nous avons comparés en particulier sur le taux d'utilisation des ressources «matériels et humains» ainsi que sur le temps d'attente de ces ressources

Mots clés : Stérilisation, Hôpital, modélisation ARIS, modèle SIMULA, Witness, Simulation.

1 Introduction

Les établissements hospitaliers sont aujourd'hui confrontés à de nombreux défis. Ils concernent notamment la réduction du temps de travail et la lutte contre les infections nosocomiales. Par ailleurs, les derniers règlements hospitaliers ont introduit des critères économiques dans les choix et les orientations des politiques hospitalières. Les objectifs qui consistent à assurer une meilleure qualité et une meilleure sécurité des soins vont désormais de pair avec des objectifs de maîtrise des dépenses de santé. Nous nous intéressons, dans cet article, à une prestation transversale dans les établissements hospitaliers : la stérilisation des dispositifs médicaux. D'une manière générale, la stérilisation des dispositifs médicaux dans les établissements de santé occupe une place primordiale dans la lutte contre l'infection. La stérilisation est la mise en oeuvre d'un ensemble de méthodes et de moyens visant à éliminer tous les micro-organismes vivants, de quelque nature que ce soit, portés par un objet parfaitement nettoyé. Le processus de stérilisation est parfois difficile à mettre en œuvre puisque les établissements doivent prendre en compte des conditions

économiques (matériel, locaux, personnel, organisation) et des conditions de faisabilité (proximité, délai, organisation, logistique). Les établissements qui ne peuvent, ou ne veulent, pas assurer la stérilisation eux-mêmes ont recours à des solutions d'externalisation [2] ; le sous-traitant qui se charge de la stérilisation peut alors être soit un autre établissement hospitalier, soit une entreprise spécialisée. Nous nous intéressons ici au cas où cette activité est réalisée en externe et notre objectif est d'étudier et d'améliorer la gestion de ce type d'activité de stérilisation.

Pour cela, nous présentons tout d'abord les procédés de l'activité de stérilisation et sa problématique [3][12], la méthodologie de modélisation [1][10] mise en oeuvre puis la modélisation de cette activité avec l'outil ARIS, ainsi que deux modèles de simulation basés sur les réseaux de files d'attente [15][4]. Le premier, implanté en SIMULA avec la classe Gpsss, suit une approche transaction alors que le second est réalisé à l'aide de l'outil de simulation, du domaine industriel, Witness qui utilise une approche station et offre des possibilités graphiques telle qu'une interface de développement et l'animation du modèle. Des résultats sont obtenus et interprétés.

2 Procédés de Stérilisation

L'externalisation encore appelée sous-traitance, désigne le fait pour une structure de se procurer des services opérationnels auprès d'un prestataire externe. La sous-traitance implique une démarche se basant sur un engagement entre deux parties contractantes sur des résultats précis. En terme plus technique, l'externalisation de la stérilisation est une opération par laquelle une entreprise dite donneur d'ordre confie à une autre entreprise, le sous-traitant, le soin d'exécuter pour elle, et selon un cahier des charges préétabli, soit la totalité ou une partie des actes de production et de service. Il s'agit donc d'un type de contrat réunissant deux partenaires : le donneur d'ordre et le sous-traitant [18]. L'externalisation offre à l'hôpital l'opportunité de se recentrer sur son métier : « garantir la meilleure sécurité des soins donnés aux patients » [16].

La stérilisation du matériel peut s'effectuer suivant différents procédés. On distingue la stérilisation à la vapeur sous pression, la stérilisation à l'air chaud et la stérilisation au gaz (principalement l'oxyde d'éthylène). Le choix du procédé de stérilisation dépend avant tout de la nature du matériel à stériliser. La stérilisation à la vapeur sous pression représente le meilleur choix. Les autres procédés sont réservés à la stérilisation du matériel pour lequel la stérilisation à la vapeur sous pression n'est pas applicable.

Le Processus de stérilisation utilise trois zones : Zone sale, Zone propre et Zone stock (Fig. 1)

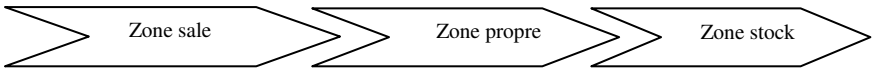


Fig. 1. Processus de stérilisation

Les dispositifs médicaux (DM) sont majoritairement des instruments réutilisables. Le processus de production de DM stériles se représente donc par un schéma cyclique appelé boucle (ou cycle de vie) de stérilisation. Ce cycle de vie comporte plusieurs étapes (Fig. 2) :

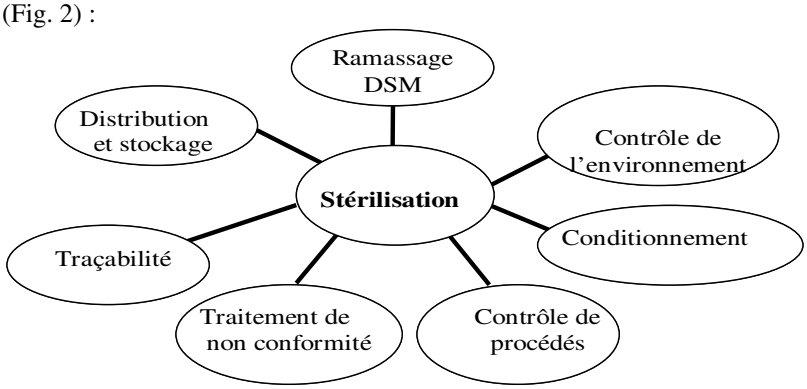


Fig. 2. Cycle de vie de la stérilisation

- *Ramassage de dispositifs médicaux (DM)* : Les armoires vont permettre le transport des boites stériles lors de la livraison, et le retour des boites non stériles. Il faut veiller à ce qu'elles soient : roulantes, légères (donc aisément mobiles), sécurisées par une fermeture hermétique renforcée, dédiées au transport soit de boites stériles, soit de boites contaminées.

- *Contrôle de l'environnement* : s'assurer de l'hygiène de l'environnement.

- *Conditionnement* : les dispositifs médicaux sont disposés dans des emballages ; il s'agit de conteneurs, de sachets ou d'emballages dans des doubles feuilles.

- *Contrôle de procédés* : Activités telles que mesurer, examiner ou essayer une ou plusieurs caractéristiques d'une entité et comparer les résultats aux exigences spécifiées en vue de déterminer si la conformité est obtenue pour chacune de ces caractéristiques.

- *Traitement de non conformité* : Une procédure de traitement de ces non conformités, prévoyant la démarche à suivre selon les problèmes rencontrés, est mise en place. Tout produit non conforme doit être exclu du circuit et une solution de rechange doit être proposée pour que le fonctionnement du bloc opératoire ne soit pas perturbé. L'absence d'un conteneur stérile ou même d'un instrument à l'intérieur d'un conteneur peut avoir des conséquences catastrophiques : désorganisation du bloc opératoire, déprogrammation d'une opération avec insatisfaction du patient et même risque vital encouru par le patient si celui-ci est déjà endormi.

- *Traçabilité* **:** Un des avantages mis en avant par l'industrie est sa traçabilité globale. La prestation repose sur une maîtrise en continu, par des enregistrements, qui permettent non seulement une traçabilité du processus, mais aussi une traçabilité de chaque instrument dans chaque boite (identification par système UNICOS). L'utilisation de l'informatique permet à l'industriel l'enregistrement par étape du DM portant un code barre. Il est donc possible de savoir pour chaque DM, si les différentes étapes du processus de stérilisation, à savoir la décontamination, le lavage, la recomposition, la stérilisation, le stockage, ont été bien réalisés. En stérilisation, il

est nécessaire de tracer : les processus, les instruments, les opérateurs, les dysfonctionnements ainsi que la maintenance, qu'elle soit préventive ou curative.

- *Distribution et stockage :* Il faut s'assurer que les locaux et équipements de stockage sont régulièrement entretenus, qu'ils sont conçus de façon à éviter toute chute ou surcharge, que la zone de stockage est spécifique aux DM stériles, que la zone de stockage est protégée de la lumière du jour, de l'humidité et des contaminations de toutes natures.

3 Méthodologie de modélisation

L'approche de modélisation que nous présentons a été proposée par Gourgand et Kellert [11], elle est basée sur la construction de deux modèles respectivement appelés modèle de connaissance et modèle d'action. Le modèle de connaissance décrit la structure et le fonctionnement du système dans un langage naturel ou graphique. Pour un système existant (étude à posteriori), le modèle de connaissance contient la connaissance acquise après observations du le système. Pour un modèle à concevoir (étude à priori), le modèle de connaissance contient les spécifications du futur système. Le modèle d'action est une traduction du modèle de connaissance dans un formalisme mathématique ou dans un langage de programmation (par exemple un langage de simulation) permettant l'évaluation des critères de performances choisis. Plusieurs modèles d'action peuvent être construits à partir du même modèle de connaissance. La construction et l'utilisation consécutives de ces deux modèles constituent le processus de modélisation.

3.1. Environnement de modélisation

Le but principal de la méthodologie de modélisation consiste à établir un modèle de connaissance aussi générique que possible et qui permet l'exécution de modèles d'action pour les systèmes spécifiques du domaine. Le modèle de connaissance demeure un modèle ouvert qui est enrichi par chaque étude de systèmes du domaine. La gestion de la connaissance et l'exécution des modèles d'action impliquent l'aide fournie par un environnement de modélisation qui est ouvert afin d'inclure de nouvelles méthodes et des outils plus efficaces. L'environnement de modélisation (Fig. 3) facilite l'échange d'informations entre les différents membres du projet et aide à la conception des modèles d'action pendant la phase d'extraction de l'information à partir du modèle de connaissance. C'est une tentative d'automatisation du processus de modélisation à l'aide de la formalisation de la connaissance, de l'analyse de données pour la détermination des caractéristiques du système, de la recherche opérationnelle et de la simulation pour l'évaluation. Les représentations graphiques et les outils d'animation permettent de vérifier le fonctionnement correct du modèle.

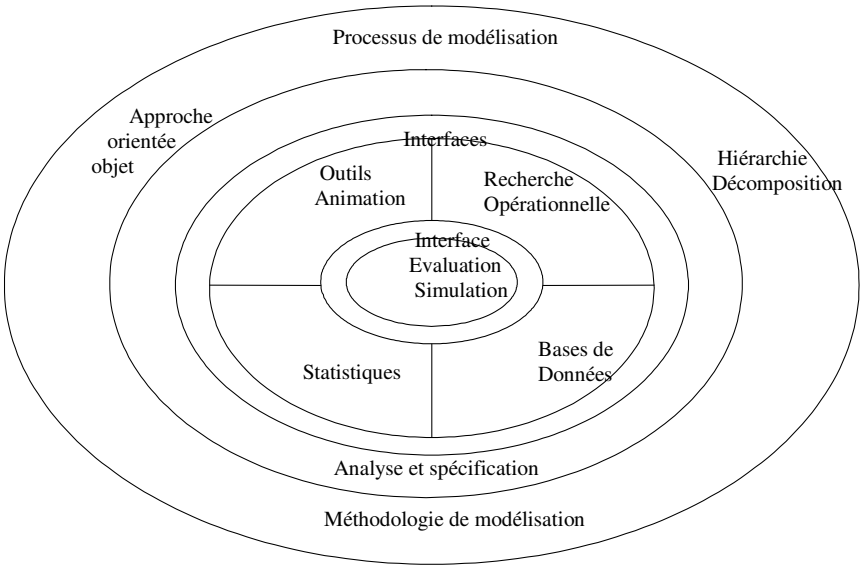


Fig. 3. Environnement de modélisation

3.2. Processus de modélisation

Le processus de modélisation est itératif (Fig. 4) [11] et se compose de quatre étapes :

- la construction d'un modèle de connaissance,
- l'élaboration d'un modèle d'action à partir du modèle de connaissance,
- l'exploitation d'un modèle d'action,
- les modifications éventuelles sur le système.

La construction du modèle de connaissance doit être réalisée en collaboration avec les experts du domaine. Ce modèle doit rester cohérent dans le temps, quel que soit son niveau de finesse et quelles que soient les évolutions apportées au système lors de son utilisation. Les objectifs à atteindre lors de l'étude doivent être clairement identifiés de façon à connaître les problèmes à résoudre. Ceci fait partie de l'identification précise du contexte et des limites de l'étude. La construction du modèle de connaissance consiste en la récolte et la formalisation de la connaissance sur le système étudié.

- Une phase d'analyse fonctionnelle et structurelle formalise le système sous une forme graphique ou syntaxique. Une méthode de décomposition peut être employée pour faciliter la formalisation de la connaissance lorsque le système étudié est complexe.

- Une phase de spécification, permet l'expression du fonctionnement des entités du système et des flux. En particulier, le système de pilotage doit être spécifié de façon précise, compte tenu des événements à prendre en compte. Le format des données d'entrée du système doit être spécifié. La spécification ainsi réalisée doit permettre aux experts en modélisation et aux experts du système de s'accorder sur le

fonctionnement (réel ou désiré) du système. La construction du modèle de connaissance nécessite l'utilisation d'outils et/ou de formalisation adaptés.

Après la construction du modèle de connaissance, il y a la traduction de ce modèle en utilisant un modèle d'action, ce modèle d'action est construit soit avec un modèle mathématique, analytique ou modèle de simulation. Enfin les performances de ce modèle d'action sont traduites en un modèle de résultat (tableau de bord,…)

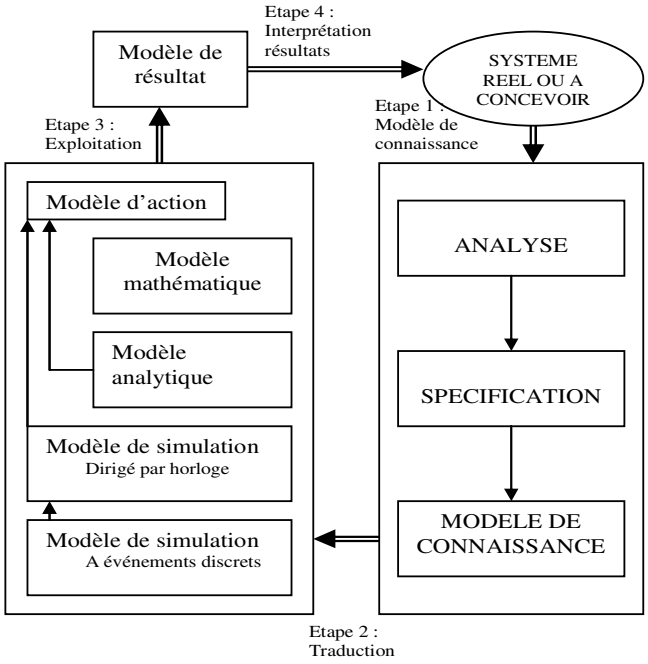


Fig. 4. Processus de modélisation

4 Mise en oeuvre et résultats

4.1 Modélisation ARIS

Le premier modèle de connaissance concernant les entités et la structure du système logistique hospitalier a été formalisé à l'aide de l'outil ARIS (Architecture of Integrated Information Systems). Cet outil, suggéré par Scheer [17], convient pour décrire des organismes, des processus et des activités [13], aussi bien que des modèles entité-association [6].

L'étude de l'organisation future de l'activité de stérilisation du NHE nous a permis de construire un modèle de connaissance selon huit niveaux repartis en trois approches (microscopique, mesoscopique et macroscopique) [5] et réalisé par l'outil ARIS (Fig. 5).

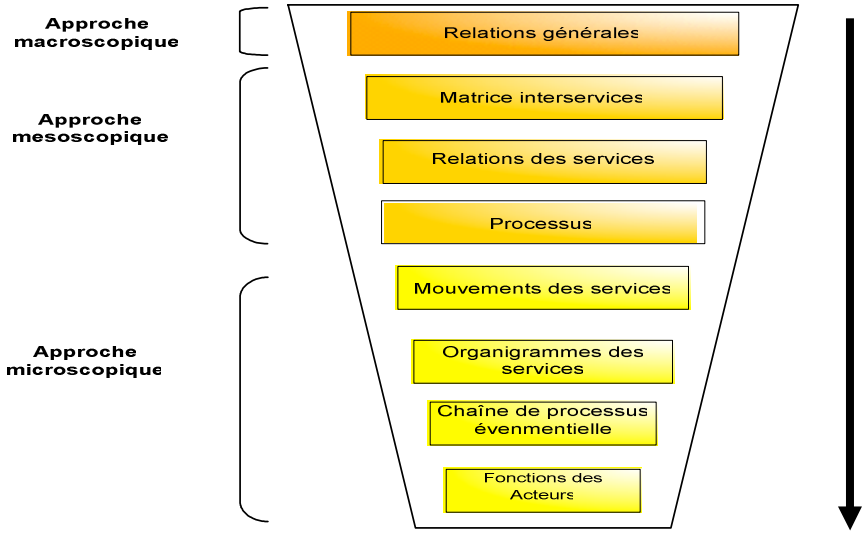


Fig. 5. Modélisation ARIS

Le fonctionnement de l'activité de la stérilisation réalisé par un diagramme de chaîne de processus événementielle est décrit par la Figure 6.

A l'arrivée des camions entre 6h et 18h les agents les déchargent et stockent les matériels dans la zone de transit en utilisant un moyen de transport. Après stockage, les matériels sont chargés sur des chariots, puis transportés vers les unités de soins, déchargés, ensuite les conditionnements vides sont chargés, transportés vers la zone de transit et enfin déchargés. Les différents sites sont Poly (Polyclinique), CHGE (Centre Hépatho-Gastro Entérologie) et UF (Unité Fonctionnelle)

4.2. Modèles d'action

Les modèles d'action réalisés sont des modèles de simulation à événements discrets. L'objectif principal est d'augmenter le taux d'utilisation des ressources jusqu'à un niveau raisonnable et de diminuer les temps d'attentes dans la zone de transit. Pour cela deux modèles sont implantés un en SIMULA et l'autre en Witness, ils sont basés sur les réseaux de files d'attente.

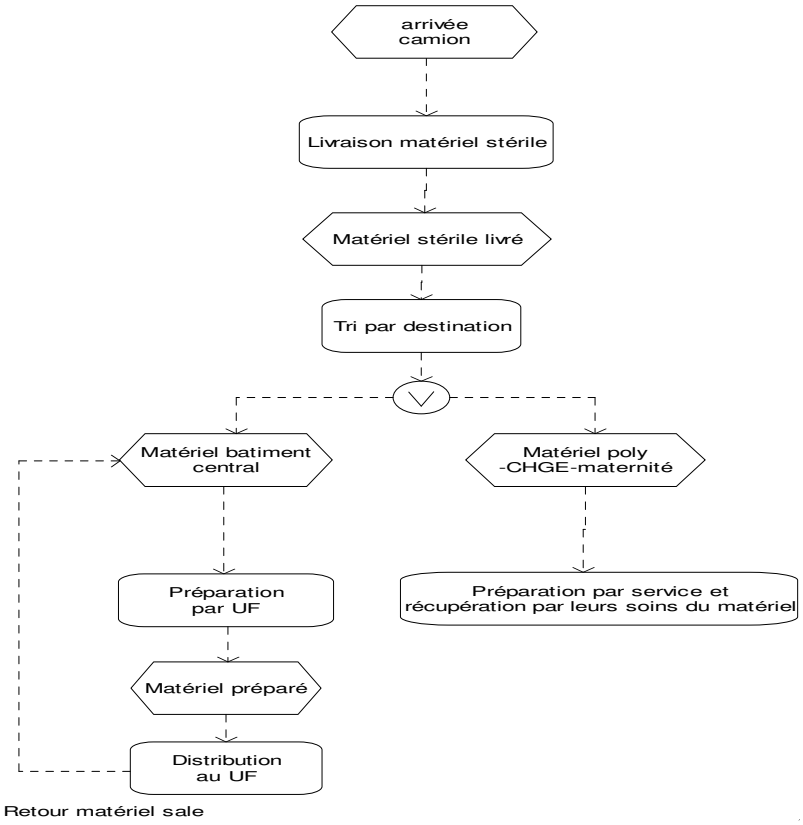


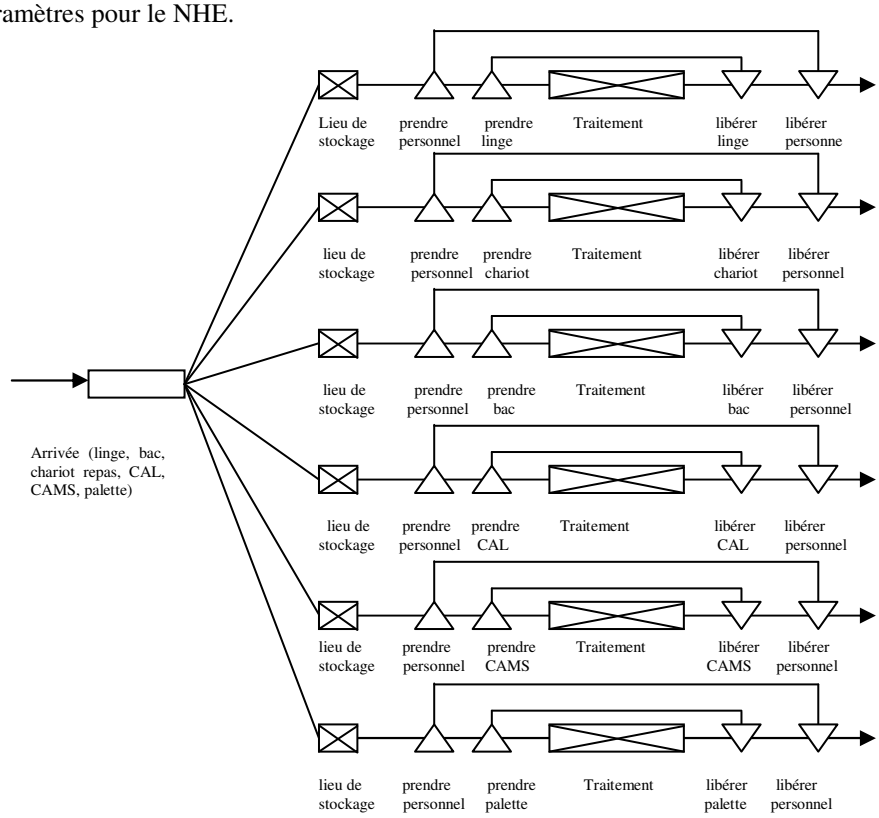
Fig. 6. Chaîne de processus événementielle d'ARIS

4.2.1. Modèle de files d'attente

La figure 7 présente un modèle de files d'attente. Ce modèle a été réalisé à partir des diagrammes de chaînes de processus événementielles d'ARIS et fonctionne comme suit : les camions arrivent à la zone de transit entre 6h et 18h. Chaque camion transporte du matériel (matériel stérile, linge, repas, pharmacie, …). Les agents distribuent les matériels aux différentes unités de soins en utilisant un moyen de transport (armoire, roller, navette,…), ou récupèrent ultérieurement les contenants vides (armoire, chariot repas, bac pharmacie,…).

La figure 7 présente les différents types de clients susceptibles d'attendre (matériel stérile, linge sale, repas, bac de pharmacie). Ces clients sont également transportés par des agents, aux moyens de deux types d'unités de flux (contenants pleins/vides) qui sont considérés dans notre cas comme des ressources. Le modèle de simulation gére des transactions (les missions correspondant au transport de matière d'un point à un autre de la zone de transite vers les unités de soins et inversement). Les flux informationnels ne sont pas transportés de la même façon donc n'interviennent pas dans le modèle proposé. Les phénomènes d'attente concernent essentiellement la non

disponibilité des agents et des moyens de transport. Pour chaque unité de flux on a représenté sa durée de stockage comme le montre le tableau 1. Le fait que la durée de traitement diffère selon les unités de soins, les moyens de transport et les unités de flux nous a incité à décomposer cette dernière en six types représentés dans le tableau 1. Nous avons utilisé des lois uniformes pour la mise au point des modèles. Une étude statistique à venir permettra de préciser les lois de probabilités et d'estimer les paramètres pour le NHE.



CAL : Centre d'approvisionnement du matériel non stérile
CAMS : Centre d'approvisionnement du matériel stérile

Fig. 7. Modèle de files d'attente

$$\text{Durée de traitement} = \text{chargement} + \text{déplacement plein} + \text{déchargement} + \text{chargement vide} + \text{déplacement vide} + \text{déchargement vide} \quad (1)$$

Tableau 1. Distribution de la durée de traitement

Unité de flux	Stockage	Chargement	Déplace-p	Décharge	Charge-v	Déplace-v	Décharge-v
Matériel stérile	U(20 ;60)	U(5 ; 10)	U(10 ;20)	U(5 ;10)	U(1 ;5)	U(3 ; 8)	U(3 ; 8)

Le tableau 2 contient les quantités pour chaque Unité de Soins (US) du NHE. La quantité totale de matériels est utilisée pour déterminer le temps de déchargement des camions (142 armoires).

Tableau 2. Distribution des quantités de stérilisation

Unité de soins	Quantité	Unité de soins	Quantité
US1	15	US7	10
US2	10	US8	15
US3	10	US9	10
US4	15	US10	05
US5	10	US11	10
US6	05	US12	10
		US13	17
			= 142

4.2.2. Modèle SIMULA

Le langage SIMULA a prouvé sa capacité à faciliter d'implantation des différentes catégories de modèles de simulation [7][8][9]. Il inclut des co-routines et des processus de la simulation à événements discrets. De nombreuses classes existantes prolongent les possibilités du langage en gestion de transaction et en calcul de statistiques. La classe de Gpsss fournit les objets de base tels que serveur, ressource, transaction, région de statistiques, rapport de simulation. Elle peut être employée comme en programmation GPSS mais avec toutes les capacités d'un langage de simulation orienté objet. Les éléments de flux (bac, palette, repas, cal, cams, linge) arrivent entre 7h et 8h (loi uniforme) selon les quantités du tableau 2, puis sont regroupés et subissent les traitements décrits dans le tableau 1. Ils sont distribués par des personnes de la zone de transit dans des unités de soins. Les ressources de transport sont représentées par des objets *storage* (ChariotL, Navette, Roller, ChariotR), ainsi que les personnes qui s'occupent de la distribution. Des régions fournissent des statistiques sur les activités des transactions : chargement, déplacement plein, déchargement, déplacement à vide, durée de séjour et d'attente. Le programme en SIMULA de l'activité logistique de stérilisation est donné en figure 9.

```
% ------------------------------------------------------------
%  Simulation logistique de Stérilisation
% ------------------------------------------------------------------
BEGIN
EXTERNAL CLASS gpsss;
Real array QS(1:13);
gpsss BEGIN
!*** Sterile;
transaction CLASS MSterile(j);integer j;!on fait une demande de mission;
    BEGIN Integer i;
      outfix(time,3,10); outtext("Arrivee MSterile"); outimage;
      outfix(time,3,10); outtext("FinDech MSterile"); outimage;
      ! Attente mission;
      Enter_storage(Personne,1); ! prise ressource personne;
      outfix(time,3,10); outtext("Mission MSterile"); outimage;
      Enter_storage(ChariotS,1);! prise ressource chariot Sterile;
      N1 := N1+1;
      Enter_region(chargementS);
```

```
      hold(uniform(5,10,u)); ! chargement;
      Leave_region(chargementS); ! fin attente;
      Enter_region(deplacementpS);
      hold(uniform(10,20,u)); ! deplacement plein;
      Leave_region(deplacementpS); ! fin attente;
      Enter_region(dechargementS);
      hold(uniform(5,10,u)); ! déchargement;
      Leave_region(dechargementS); ! fin attente;
      Enter_region(chargementV);
      hold(uniform(1,5,u)); ! chargement vide;
      Leave_region(chargementV); ! fin attente;
      Enter_region(deplacementvS);
      hold(uniform(3,8,u)); ! déplacement vide;
      Leave_region(deplacementvS); ! fin attente;
      Enter_region(dechargementV);
      hold(uniform(3,8,u)); ! déchargement vide;
      Leave_region(dechargementV); ! fin attente;
      Leave_storage(chariotS,1); ! libérer chariot Sterile;
      Leave_storage(Personne,1); ! libérer personne;
      N1 := N1-1;
      outfix(time,3,10); outtext("Fin MSterile"); outimage;
    END;
% ----------------------------------------------
```

Fig. 9. Programme SIMULA de l'activité de stérilisation

Le tableau 4 montre que le personnel (4 agents) a un taux d'occupation de 97.61% sans tenir compte des pauses pour les 8 heures de présence.

Tableau 4. Résultats du modèle SIMULA

```
   * storages *
   ************
                          avg.    avg.time   contents
                 entries contents  transit   now   max capacity   util.

   chariotS           13     1.22    44.98     0     2        2    60.90%
   personne           13     2.26    83.35     0     4        4    56.43%


   * regions *
   ***********
                          avg.    avg.time   contents  non-zero   zero
                 entries contents  transit   now   max  transit  entries

   chargementS        13     0.20     7.45     0     2     7.45       0
   deplacementpS      13     0.41    15.13     0     2    15.13       0
   dechargementS      13     0.22     8.18     0     2     8.18       0
   deplacementvS      13     0.15     5.34     0     2     5.34       0
   chargementv        13     0.09     3.22     0     2     3.22       0
   dechargementv      13     0.15     5.66     0     2     5.66       0
```

4.2.3. Modèle Witness

Quelques parties du système hospitalier sont spécifiées avec le langage UML pour compléter l'analyse et spécifier plus précisément les processus. Un module supplémentaire de simulation est disponible pour ARIS, mais pour les besoins graphiques et surtout d'animation 3D, l'outil de simulation Witness a été préféré, dans un premier temps, afin de concevoir et mettre en application des modèles d'action[14].

Le modèle Witness (Fig. 9) fournit une représentation graphique (une interface) animée de la stérilisation. Il utilise des articles comme le matériel stérile (palette, chariot repas, bac, CAL, CAMS, sac de linge) avec les quantités du tableau 2 et les moyens de transport : chariot linge, chariot repas, roller et navette. Le routage de ces articles se fait par des machines M et MUS qui représentent le début de leur périple et les unités de soins où ils sont distribués selon les probabilités utilisées dans le tableau 1.. Le stock vide est utilisé pour stocker les contenants vides. La durée du traitement appliqué aux articles depuis la zone de transit jusqu'aux unités de soins (US) est donnée sous la forme de la formule (1).

Le tableau 5 et la figure 10 représentent les résultats du modèle Witness, ils portent sur le transport des articles : nombre de contenants en entrées (Entrée) et le temps moyen de traitement (Tpsm), ainsi que sur la ressource agent (opérateur personne) : taux d'occupation (%Occup), nombre des tâches commencées (Comm), nombre des tâches finies (Fini) et le temps moyen d'utilisation (Tpsm). Le taux d'occupation du personnel est compatible (45%) avec celui obtenu avec le modèle SIMULA (56%). Une étude par réplications du modèle SIMULA permettrait de préciser l'intervalle de confiance.

D'après le tableau 5, nous remarquons que le taux d'utilisation du personnel de la zone de transit est moyennement faible pour un effectif de 4 personnes, par rapport au résultat fait par SIMULA et cela est du au manque de données fiables et réelle. Nous attendons des estimations de données statistiques plus réalistes pour le NHE.

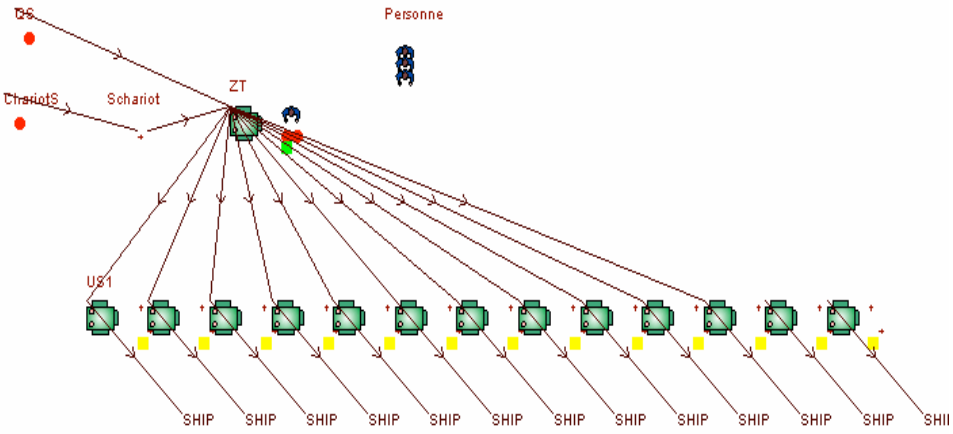


Fig. 9. Modèle Witness

Tableau 5. Résultats Witness

ARTICLES		RESSOURCES	
Critère	ChariotS	Critère	Personne
Entrée	13	%Occup.	45.20
Tpsm	25.59	Comm.	41
		Fini.	39
		Tpsm	19.47

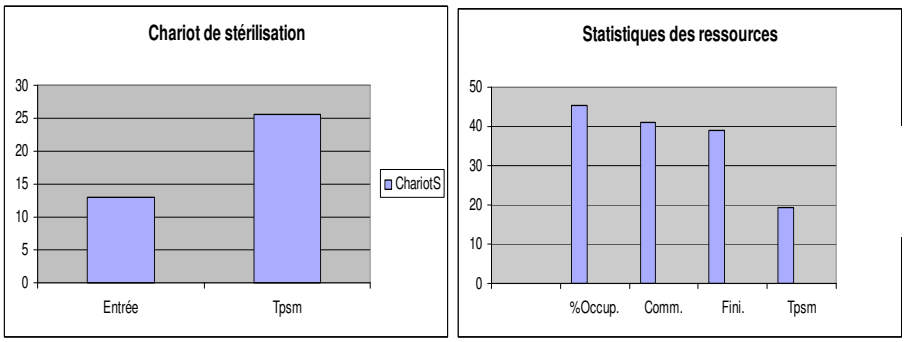


Fig. 10. Graphes des résultats Witness

5 Conclusion

L'externalisation de la stérilisation semble une solution d'avenir pour les établissements de santé. La stérilisation est une activité qui est devenue très spécifique: elle doit répondre à des normes de qualité et à une réglementation contraignante. Pour cela nous avons présenté l'activité de la stérilisation et l'utilité du modèle de connaissance spécifié avec l'outil ARIS, ainsi que le passage de ce modèle vers un modèle d'action ou de simulation implanté en SIMULA et avec Witness . Le modèle en SIMULA et celui en Witness ont fourni des résultats. Les résultats dans SIMULA sont plus proches de la réalité que dans Witness. Le taux d'utilisation des ressources est moyennement faible pour un effectif de 4 personnes. Nous attendons des estimations de données statistiques plus réalistes pour le NHE.

References

1. André, V. : Modélisation et simulation des flux logistiques du nouvel hôpital d'Estaing. Rapport technique, Université d'Auvergne, Clermont-Ferrand, octobre (2005)
2. Bardet, E. : Externalisation de la stérilisation : conditions de mise en oeuvre, avantages, inconvénients. Mémoire ENSP (Ecole Nationale de la Santé Publique) de Pharmacien Inspecteur de la Santé Publique. Rennes, France, (2003), 32p

3. Benanteur, Y., Rollinger, R., Saillour, J.-L. : L'organisation logistique et technique à l'hôpital. Éditions ENSP, (2000), 185p
4. Chabrol, M., Fénièse, P., Gourgand, M., Tchernev, N. : Un environnement de modélisation pour le système d'information de la suply chain. Application au Nouvel Hôpital d'Estaing, Clermont-Ferrand, (2005)
5. Chauvet, J., Dessommes, N., Durand, N., Quinkal, A. : Un modèle de connaissance pour le nouvel hôpital d'Estaing. DESS management, Université d'Auvergne, Clermont-Ferrand, mars (2005)
6. Chen, P.: The entity relationship model - Toward a unified view of data. ACM Transaction on data base system, Vol. 1(1), mars (1976)
7. Clymer, J.R.: System analysis using simulation and Markov Models. Prentice Hall. (1990A)
8. Clymer, J.R., Corey, P.D., Nili, N.: Operationnal evaluation modelling. Simulation, Vol. 55(6), décembre (1990B), 327-336
9. Clymer, J.R., Hernandez, D.: OPEM distributed simulation. Simulation, Vol. 57(6), Décembre (1991), 395-404
10. Combes, C. : Un environnement de modélisation pour les systèmes hospitaliers. Thèse de doctorat, Université Blaise Pascal, Clermont-Ferrand, 27 Octobre (1994)
11. Gourgand, M., Kellert, P. : Conception d'un environnement de modélisation des systèmes de production. 3ème congrès international de Génie Industriel, Tours, (1991)
12. Gourgand, M., Mebrek, F., Tanguy, A.: Hospital logistic modelling and simulation case study: brancardage. ESM05, Porto, Portugal, 24-26 october (2005)
13. Green, P., Roseman, M.: Modelling : An ontological evaluation. In Information systems, ated process, Vol. 25. (2000)
14. Lanner group. : Formation Witness. Edition Anglettre, (2003)
15. Mebrek, F., Tanguy, A. : Modélisation et simulation à événements discrets du pôle imagerie d'un hôpital moderne. MOSIM'06, Maroc-Rabat, 3-5 avril (2006)
16. Navarro, I., Rodrigues, M. : La sous-traitance en pharmacie, application élaboration d'un contrat de sous-traitance en stérilisation. Mémoire UV gestion hospitalière, DESS de pharmacie hospitalière et des collectivités. Faculté de pharmacie de Montpellier, mai (2000)
17. Scheer, A.W. : ARIS - Business Process Modelling. Springer. (2002)
18. Viala, G. : La sous-traitance dans l'industrie pharmaceutique, aspects juridiques, STP Pharma Pratiques, Vol. 3(6) (1993), 453-462

Towards semantic interoperability: An ontology mapping approach

Afaf Merazi[1], Sidi Mohamed Benslimane[2] and Mimoun Malki[1]

[1] EEDIS Laboratory, University of of Sidi Bel Abbes, Algeria
Afaf.merazi@gmail, Malki@univ-sba.dz
[2] LIRIS Laboratory, University of Claude Bernard, France
Sidi-mohamed.benslimane@liris.cnrs.fr

Abstract. Ontology is increasingly seen as a key factor for automatic processing of machines, and interoperability between heterogeneous information systems. Ontology mapping is becoming a crucial aspect in providing the background knowledge required for solving heterogeneity problems between semantically described data sources, and accessing distributed information repositories. Developing such ontology mapping has been a core issue of recent ontology research. In this paper we present a three-layer framework to (semi-)automatically discovering ontology mapping. We show how such resulting mapping is used for resolving semantic interrogation tasks, and enabling runtime semantic interoperability across heterogeneous information systems using semantic web technologies.

Keywords: Ontology mapping, Semantic Web, Semantic interoperability, Similarity measures.

1. Introduction

Intuitively, ontologies can be seen as defining the basic terms and relations of a domain of interest, as well as the rules for combining these terms and relations [1]. They are considered to be an important technology for the semantic Web. Ontologies are used for communication between people and organizations by providing a common terminology over a domain. They provide the basis for interoperability between systems. They can be used for making the content in information sources explicit and serve as an index to a repository of information. Further, they can be used as a basis for integration of information sources and as a query model for information sources. They also support clearly separating domain knowledge from application-based knowledge as well as validation of data sources. Many onthologies have already been developed and many of these ontologies contain overlapping information. Often we would therefore want to be able to use multiple ontologies. For instance, companies may want to use community standard ontologies and use them together with company-specific ontologies. Applications may need to use ontologies from different areas or from different views on one area. Ontology builders may want

to use already existing ontologies as the basis for the creation of new ontologies by extending the existing ontologies or by combining knowledge from different smaller ontologies. So to bring together these disparate source ontologies, two approaches are possible: (1) merging the ontologies to create a single coherent ontology, or (2) aligning the ontologies by establishing links between them and allowing the aligned ontologies to reuse information from one another. Ontology merging is the process of generating a single, coherent ontology from two or more existing and different ontologies related to the same subject [2]. A merged single coherent ontology includes information from all source ontologies but is more or less unchanged. The original ontologies have similar or overlapping domains but they are unique and not revisions of the same ontology [3]. Ontology alignment is the task of creating links between two original ontologies. Ontology alignment is made if the sources become consistent with each other but are kept separate [4]. Ontology alignment is made when they usually have complementary domains.

Until now, ontology designers have performed this complicated process of ontology merging and alignment mostly by hand, without any tools to automate the process or to provide a specialized interface. It is unrealistic to hope that merging or alignment at the semantic level could be performed completely automatically. It is, however, possible to use a hybrid approach where a system performs selected operations automatically and suggests other likely points of alignment or merging. Some researchers view the mapping process as an integral part of alignment or merging. Clearly, mapping is an essential aspect of alignment and could also be used to initiate merging.

In this paper we present a three-layer framework to (semi-)automatically discovering ontology mapping. We show how such resulting mapping is used for resolving semantic interrogation tasks, and enabling interoperability across heterogeneous information systems using semantic web technologies. The rest of the paper is organized as follows. Section 2 introduces the overall of our framework. Section 3 describes the mapping discovery and representation layer. Section 4 explains some features of the semantic query layer. Section 5 presents the implementation of our system and details some experimental result. Finally, Section 6 contains concluding remarks and suggests some future works.

2. The overall framework

The overall system architecture, as shown in figure 1, involves the following layers:

The information source layer includes a set of data sources which typically store their data in relational databases. Each data source is related to a local ontology by semantic relationship. Data sources schemas are mapped into corresponding classes or properties defined in the local ontology.

At the mapping discovery layer, a mapping generation process produces semantic correspondences between the local ontologies. The discovered mapping is encoded using standardized mapping representation languages.

At the semantic query layer a form-based query interface is offered to construct semantic queries over the set of ontology mapping. A semantic query is automatically

generated at runtime, and submitted to the semantic query engine, where the query will be rewritten into a set of SQL queries using mapping contained in the ontology mapping registry. Finally, the results of SQL queries will be merged and forwarded back to the end-user.

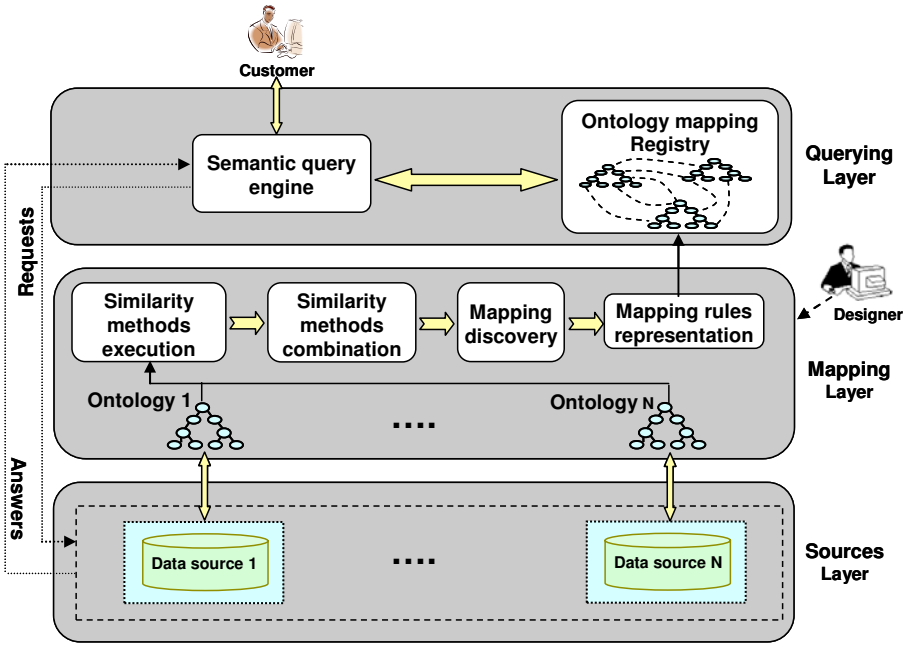


Fig. 1. Ontology mapping and semantic querying framework

3. Ontology mapping layer

Ontology mapping is one of the core tasks for ontology interoperability. A systematic and epistemological account of ontology mapping and its related definitions is provided in the surveys of [5] for ontology mapping, [6] for database schema matching, and [7] for combined views of these two mapping regimes. Due to the wide range of expressions used in this area (merging, alignment, integration etc.), we adopt the following definition for the term mapping: "Given two ontologies O_1 and O_2, mapping one ontology onto another means that for each entity (concept C, relation R, or instance I) in the source ontology O_1, we find a corresponding entity, which has the same intended meaning, in the target ontology O_2" [8].

The mapping generation methods basically operate in three phases. Firstly, given two ontologies, how do we find the similarities between them and how do we determine which concepts and properties represent similar notions. Second, according to the similarity measures obtained in the first step, the methods specify how to represent the mapping between entities, and finally the resulting mapping is used for various interrogation tasks (e.g. query answering, web-service composition, etc.)

3.1 Mapping discovery

The task of finding mapping has been an active area of research in both database and ontology communities [5, 6]. We identify two major architectures for mapping discovery between ontologies. For the first approach, finding correspondences between two extensions can be facilitated by a general upper ontology [9]. The second set of approaches comprises heuristics-based or machine learning techniques that use various characteristics of ontologies, such as their structure, definitions of concepts, and instances of classes, to find mapping. To compare two entities from two different ontologies, one considers their characteristics, i.e. their features. The features of ontological entities are extracted from extensional and intentional ontology definitions [8].

Each of the features can be used to calculate ontology mapping function between the entities of the ontologies to be mapped.

Definition 1. Formally we define an ontology mapping function, Map: $O_1 \rightarrow O_2$
$Map(e_{O1}) = e_{O2}$, if $Sim(e_{O1}, e_{O2}) >$ threshold.
Where O_i: ontology, e_{Oi} : entities of O_i, $Sim(e_{O1}, e_{O2})$: similarity function between two entities e_{O1} and e_{O2}.

The ontology mapping function is based on different similarity measures.

Definition 2. A similarity measure is a real-valued function $sim(x, y)$: $O \times O \rightarrow [0, 1]$ measuring the degree of similarity between x and y.
$Sim(x, y) = 1 \rightarrow x = y$: two entities are identical.
$Sim(x, y) = 0$: two entities are different and have no common characteristics.

3.1.1 Lexical-based mapping discovery. The more evident method for relating two entities is to look at the label describing them. This label generally consists of one term that has to be compared with the labels of the other ontology. Several ideas have already been created to compare to stings.

String distance. String distance or string similarity algorithms take as an input two string and return a value indicating the distance or similarity between them.

Rule 1. *Two entities are identical if they have the same identifier (e.g. URIs) or the same name (e.g. RDF label).*

For two concepts e_1 and e_2, the string similarity measure Sim_{stri} can be given based on edit distance, which is shown in formula (1).

$$Sim_{stri}(e_1, e_2) = \max\left(0, \frac{\min(|e_1|, |e_2|) - ed(e_1, e_2)}{\min(|e_1|, |e_2|)}\right) \in [0,1] \quad \textbf{(1)}$$

Where ed is the edit distance formulated by Levenshtein [10]. It measures the minimum number of token insertions, deletions, and substitutions required to transform one string into another.

Lexical semantics. The lexical semantics similarity measure explores the semantic meanings of the word constituents by using external resources, like user-defined lexica and/or dictionaries (e.g. Wordnet, [11]) to help identify synonyms in matching.

Rule 2. *Two entities are identical if they are synonyms.*

For two concepts e_1 and e_2, the lexical semantics similarity measure Sim_{lsem} can be given using the WordNet synsets (i.e., term for a sense or a meaning by a group of synonyms) based on the formula (2).

$$Sim_{lsem}(e_1, e_2) = 1/length(e_1, e_2) \quad \textbf{(2)}$$

Where *length* is the length of the shortest path between two entities using node-counting. Based on formula (1) and (2), we give the lexical similarity measure as formula (3).

$$Sim_{lexi}(e_1, e_2) = \max\left[Sim_{stri}(e_1, e_2), Sim_{lsem}(e_1, e_2)\right] \quad \textbf{(3)}$$

3.1.2 Structure-based mapping discovery. Lexical similarity measure is not sufficient. This section presents how the concept's structure and property's structure, helps in finding similarities between ontologies. A concept's structure includes its properties and relations. A property's structure includes its domain, range, and constraints. The structure based similarity of two concepts is defined by the combined similarity between the two concept's structures.

Rule 3. *Two concepts are equal if their properties are equal.*

The structural similarity about two concepts e_1 and e_2 is denoted as Sim_{stru} which is shown as formula (4).

$$Sim_{stru}(e_1, e_2) = \frac{\sum_{i=1}^{n}\left(\sum_{j=1}^{m} Sim_{prop}\left(e_{1p_i}, e_{2p_j}\right)\right)}{(n+m)/2} \quad \textbf{(4)}$$

Where p_i and p_j are the properties of e_1 and e_2 respectively n and m are the properties number of e_1 and e_2 respectively.

Rule 4. *Two properties are equal if they have the same name (using Rule 1 and Rule 2), or if the domain and range (the original and the result concept) of two properties are equal.*

The structure similarity about two properties p_i and p_j of respectively the two concepts e_1 and e_2 is denoted as Sim_{prop} which is shown as formula (5).

$$Sim_{prop}\left(e_{1p_i}, e_{2p_j}\right) = \max\left(Sim_{lexi}(e_{1p_i}, e_{2p_j}), \frac{Sim(e_{1p_i^{dom}}, e_{2p_j^{dom}}) + Sim(e_{1p_i^{ran}}, e_{2p_j^{ran}})}{2}\right) \quad \textbf{(5)}$$

Where p_i^{dom}, p_i^{ran} and p_j^{dom}, p_j^{ran} are the domain and the range of the proprieties p_i and p_j respectively.

3.1.3 Taxonomy-based mapping discovery. It is tempting to use back-propagation in the inheritance hierarchy, as we did for structure matching. That is, if two concepts have similar specializations (generalization), the concepts are probably related. Given two concepts, we estimate their similarity by comparing the similarities of their ascendants (descendants).

Rule 5. *Two concepts are similar if their sub-concepts are the same.*

The similarity about sub-concepts of two concepts e_1 and e_2 is denoted as Sim_{subc} which is shown as formula (6).

$$Sim_{subc}(e_1, e_2) = \frac{\sum_{i=1}^{l}\left(\sum_{j=1}^{k} Sim\left(e_{1d_i}, e_{2d_j}\right)\right)}{(l+k)/2} \quad \textbf{(6)}$$

Where d_i and d_j are the sub-concepts of e_1 and e_2 respectively l and k are the sub-concepts number of e_1 and e_2 respectively.

Rule 6. *Two concepts are similar if their super-concepts are the same.*

The similarity about super-concepts of two concepts e_1 and e_2 is denoted as Sim_{supc} which is shown as formula (7).

$$Sim_{sup\,c}(e_1, e_2) = \frac{\sum_{i-1}^{l}\left(\sum_{j-1}^{k} Sim\left(e_{1a_i}, e_{2a_j}\right)\right)}{(l+k)/2} \quad \textbf{(7)}$$

Where a_i and a_j are the sub-concepts of e_1 and e_2 respectively l and k are the sub-concepts number of e_1 and e_2 respectively. Based on formula (6) and (7), we give the conceptual similarity measure as formula (8).

$$Sim_{taxo}(e_1,e_2)=\left[W_{subc} * Sim_{subc}(e_1,e_2)+W_{supc} * Sim_{supc}(e_1,e_2)\right] \quad \textbf{(8)}$$

With W_{subc} and W_{supc} being the weights which indicates respectively the importance of the similarity methods Sim_{subc} and Sim_{supc}, such as $W_{subc} + W_{supc}$=1.00

3.1.4 Similarities aggregation. To achieve high accuracy for a large variety of ontologies, a single similarity method may be unlikely to be successful. Hence, combining different similarity methods is an effective way. A combination of the presented rules leads to better mapping results compared to using only one at a time.

A general formula (9) can be given by the following average over the weighted similarity methods.

$$Sim(e_1,e_2)=\left[W_{lexi} * Sim_{lexi}(e_1,e_2)\right]+\left[W_{stru} * Sim_{stru}(e_1,e_2)\right]+\left[W_{taxo} * Sim_{taxo}(e_1,e_2)\right] \quad \textbf{(9)}$$

With W_{lexi}, W_{stru}, and W_{taxo} being the weights indicating respectively the importance of the similarity methods Sim_{lexi}, Sim_{stru}, and Sim_{taxo}, such as: $W_{lexi}+ W_{stru}+ W_{taxo}$ =1.00

3.1.5 Process. We briefly introduce a canonical process that subsumes all the mapping discovery steps illustrated in Figure 1. It is started with two ontologies, which are going to be mapped onto one another, as its input.

Similarity methods execution. A main step during mapping discovery is the execution of multiple independent mapping algorithms based on all the introduced similarity measures (Rule 1 through Rule 9). Each algorithm determines similarity values between candidates mapping (e_1, e_2) based on their definitions in O_1, O_2, respectively. Each mapping algorithm determines an intermediate mapping result according to the similarity value between 0 and 1 for each possible candidate mapping. The result of the mapping execution phase with k algorithms, m entities in O_1 and n entities in O_2 is a $k * m * n$ cube of similarity values, which is stored in the repository for later strategies combination steps.

Similarity methods combination. By multiple mapping algorithms, there are several similarity values for a candidate mapping (e_1, e_2). This step is to derive the combined mapping result from the individual algorithm results stored in the similarity cube. For each combination of ontology entities the algorithm-specific similarity values are aggregated into a combined similarity value, e.g. by taking the average or maximum value.

Mapping discovery. This step uses the individual or combined similarity values to derive mapping between entities from O_1 to O_2. Some mechanisms here are, using thresholds or maximum values for similarity mapping, performing relaxation labeling [12], or other criteria. The mapping process supports an optional designer interaction phase for mapping correction. The designer's specified mapping will influence the similarity computations for the neighborhood of the respective entities and thus can improve the mapping accuracy of structural algorithms. Eventually, the output is a mapping table including multiple entries of $Map(e_1, e_2)$ from O_1 to O_2.

3.2 Mapping representation

The discovered mapping is encoded using standardized mapping representation languages [13]. The language provide a formalism for describing a mapping element as a 5-uple: (id, e_1, e_2, n, R), where id is a unique identifier of the given mapping element; e_1 and e_2 are the entities (Concepts, properties, relations) of the first and the second schema/ontology respectively; n is a similarity measure in some mathematical structure (typically in the [0,1] range) holding for the correspondence between the entities e_1 and e_2; R is a relation holding between the entities e_1 and e_2. The relation R is defined based on the confidence measure. If $Sim(e_1, e_2) = 1$ then R is equivalence (=) relation. If $Sim(e_1, e_2) = 0$ then R is disjointness ($\perp$) relation. If $Sim(e_1, e_2) > t$ (threshold) then R is subsume ($\supseteq$) relation.

To illustrate our mapping discovery and representation process, we take as input a set of simplified ontologies concerning travel, accommodation and attraction sites. The ontologies share some related concepts and relations. The execution of multiple

mapping discovery algorithms described above, will generate the e-Tourism ontology describing tourism domain described by a set of ontology mapping (see Figure 2).

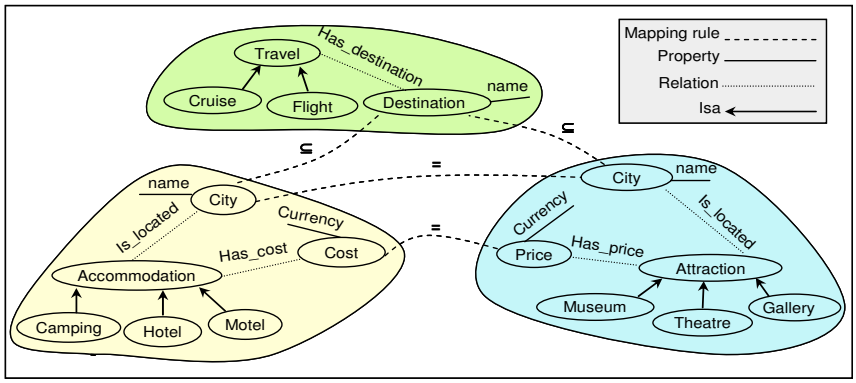


Fig. 2. Example of simplified e-tourism ontology

The mapping rules are stored in RDF format. For example, cost of the accommodation ontology is equivalent to price of the attraction ontology. This mapping rule is formulated as follow:

```
<map>
<entity1 rdf:resource='http://www.example.org/Accommodation/ontology#Cost'/>
<entity2 rdf:resource='http://www.example.org/Attraction/ontology#Price'/>
<measure rdf:datatype='&xsd;float'>1.00</measure>
<relation> = </relation>
</map>
```

4. The semantic query layer

We select SPARQL [14] as our query language which is declarative, expressive and efficient for repositories based on RDF graph (compatible with OWL). In SPARQL, an SQL-like query describes desired graph patterns with some variables and the selected boundable variables are selected as query result.

At the query layer, semantic query engine is used to process semantic SPARQL queries. Firstly, it gets mapping information from the semantic ontology mapping registry to decompose the semantic SPARQL query into a set of SPARQL sub-queries and to translate them to SQL sub-queries. Afterward, the SQL sub-queries are dispatched toward the specific databases. Finally, the results of SQL sub-queries will be recomposed and transformed back to semantically-enriched format in a way that the user indicates.

Figure 3 illustrates a query expressed in SPARQL. The query allows selecting the hotels in 'Paris' that have a cost lower than 60 euros, and displaying all its museums. The original query is translated into a set of SPARQL sub-queries (One query about accommodation, and one query about attraction).

Semantic SPARQL Query

```
PREFIX  trv: <owlmapping.example.com/Travel/ontology#>
PREFIX acc: <owlmapping.example.com/Accommodation/ontology#>
PREFIX att: <owlmapping.example.com/Attraction/ontology#>
SELECT ?myDest ?hotel ?hprice ?museum
FROM <owlmapping.example.com/Travel/ontology>
FROM <owlmapping.example.com/Accommodation/ontology>
FROM <owlmapping.example.com/Attraction/ontology>
WHERE {_:anyDest trv:name  ?myDest .
FILTER ( ?myDest =”Paris”).
?hotel acc:Hotel .
?hcity acc:name ?myDest .
?hotel acc:is_located ?hcity .
?hotel acc:hasCost ?hcost .
FILTER (?hcost < 60 ) .
?museum a att:Museum .
?mcity att:name ?myDest .
?museum att :is_located ?mcity }
```

Query to Accommodation Ontology

```
PREFIX acc: <owlmapping.example.com
        /Accommodation/ontology#>
SELECT ?myDest ?hotel ?hprice
FROM <owlmapping.example.com
       /Accommodation/ontology>
WHERE {
  ?hcity acc:name ?myDest
 FILTER ( ?myDest = ”Paris”)
 ?hotel a acc:Hotel
 ?hotel acc:is_located ?hcity
 ?hotel acc:hasCost ?hprice
 FILTER ( ?hprice < 60 ) }
```

Query to Attraction Ontology

```
PREFIX att: <owlmapping.example.com
             /Attraction/ontology#>
SELECT ?myDest ?museum
FROM <owlmapping.example.com
       /Attraction/ontology>
WHERE {
 ?mcity att:name ?myDest
 FILTER ( ?myDest = ”Paris”)
 ?museum a att:Museum
 ?museum att :is_located ?mcity }
```

Fig. 3. Semantic query expressed in SPARQL and its decomposition

5. Implementation and experimental results

In this section, we present some experiments we performed to assess the effectiveness of the ontology mapping development rules, and to verify that the proposed approach can contribute to help users resolving semantic interrogation tasks.

5.1 Prototype

A prototype is developed using Java (j2sdk 1.4.2), OWL-API [15], and Java WordNet Library (jwnl) for ontology mapping discovery and semantic query processing. The prototype that has a user-friendly GUI (Figure 4), has been implemented in order to experiment and verify that the proposed approach is doable.

To perform the ontology mapping discovery process, some parameters such as the mapped ontologies, information for the similarity methods combination, threshold, and weights, are given in an input configuration. The output ontology mapping can be formalized in the following standard formats: OWL, RDF/XML, SEKT Mapping, SWRL [16], and XSLT [17].

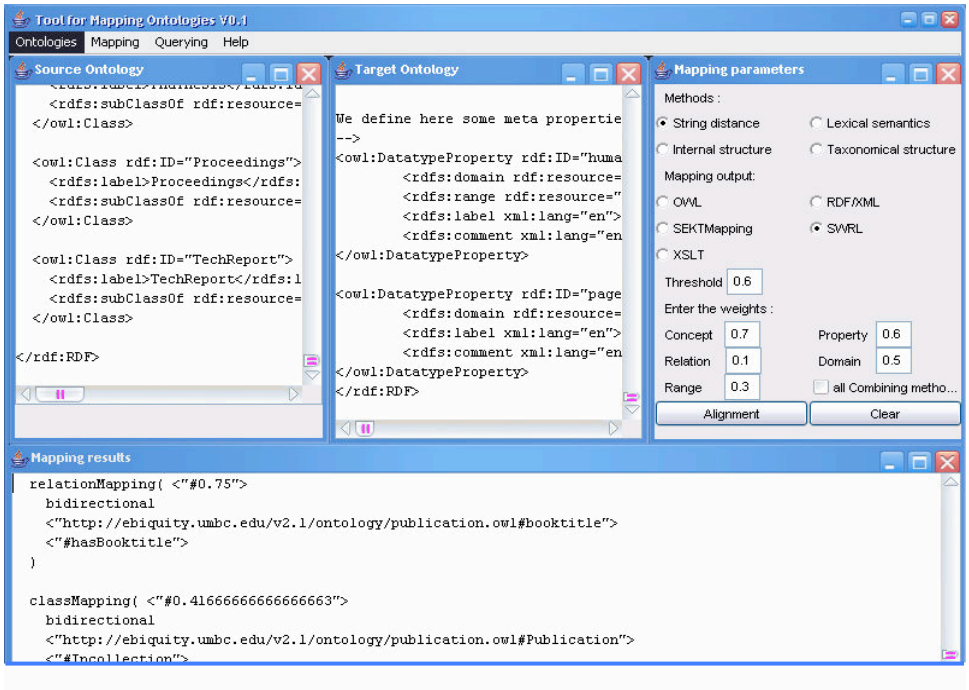


Fig. 4. Snapshot of the ontology mapping discovery tool.

5.2 Empirical Experiment

We evaluated our algorithm on the benchmark ontologies from the Ontology Alignment Evaluation Initiative (OAEI 2005, [18]). The benchmark ontologies consist of versions of a base ontology, where different aspects have been changed. For most of the following tests, we focus on six interesting ontologies: In two cases (ontologies 10 and 227 from the test suite), all names and labels have been replaced with synonyms or foreign words, and in four cases, independently developed "real-world" ontologies that describe the same domain have been used (304, 615, 997, 1072). We tested various techniques to evaluate the effectiveness of our algorithm using information retrieval metrics such as precision, recall and F-measure [19]. The results in figure 5 show that the configuration that combines different similarity methods performs better or equal than the configuration with only one similarity method.

To find out what value for the threshold is best, we took a closer look at ontologies 615. Figure 6 shows the relation between the threshold and the precision, recall and F-measure on ontologies 615. In this ontology, precision and recall at threshold 0 are lower. When raising the threshold, recall drops only slightly while precision increases rather quickly. Maximum F-measure is reached at a threshold between 0.7 and 0.85. We have to conclude that the best cut-off value for our mapping algorithm depends strongly on the dataset.

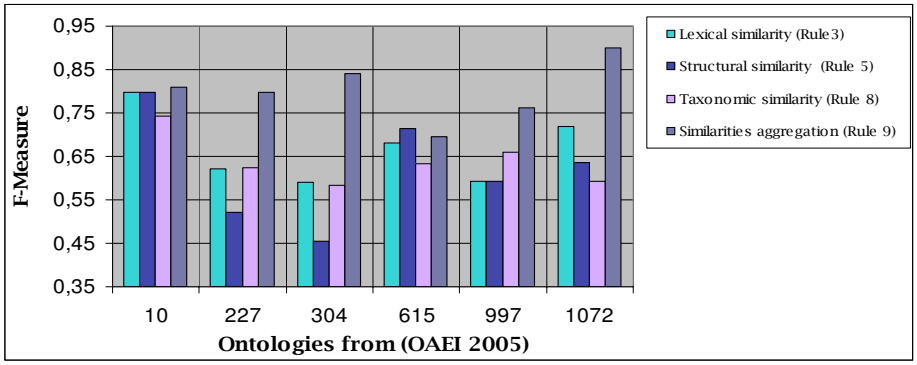


Fig 5: Comparison of different configurations of our algorithm

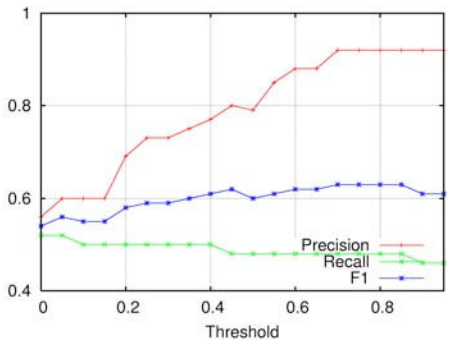


Fig 6: Relation between threshold and precision, recall and F1 for ontologies 615.

6. Conclusions

The increasing popularity of the semantic Web poses new challenges for ontology mapping. If we accept that mapping ontologies can provide a better knowledge management of the heterogeneous sources on the semantic Web, then issues of inconsistency and incompleteness need to be addressed. Therefore ontology mapping systems that operate in this environment should have the appropriate mechanisms to cope with these issues. In this paper, we have presented a three-layer framework to (semi-)automatically resolving semantic interrogation tasks, and enabling runtime semantic interoperability across heterogeneous information systems using semantic web technologies. We have implemented a tool to support this task and presented some experimental results.

As a perspective of our work, we plain to treat the maintenance of the mappings: if some changes occur in the local ontologies, some of the mappings may become inconsistent; the problem is therefore to detect the inconsistent mappings and to propagate the changes into the mapping definitions.

References

1. Gómez-Pérez A. Ontological Engineering: A state of the Art. Expert Update. British Computer Society, 2(3):33-43, 1999.
2. H. Sofia Pinto, A. Gomez-Perez, J. P. Martins, "Some Issues on Ontology Integration", In Proc. Of IJCAI99's Workshop on Ontologies and Problem Solving Methods: Lessons Learned and Future Trends, 1999.
3 Helena Sofia Pinto, Joao P. Martins, "A Methodology for Ontology Integration", Proceedings of the International Conference on Knowledge Capture, Technical papers, ACM Press, pp. 131-138, 2001.
4 N. Noy and M. Musen, "PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment." Proceedings of the National Conference on Artificial Intelligence (AAAI), 2000.
5. Kalfoglou, Y., and Schorlemmer, W. M., Ontology mapping: the state of the art. The Knowledge Engineering Review, 18(1):1–31, 2003.
6 Rahm, E., and Bernstein, P. A., A survey of approaches to automatic schema matching. VLDB J., 10(4):334–350, 2001.
7. Noy, N.F., Semantic integration: A survey of ontology-base approaches. SIGMOD Record, 33(4), 2004.
8. Ehrig, M., and Sure, Y., Ontology mapping - an integrated approach. In First European Semantic Web Symposium, ESWS, Lecture Notes in Computer Science, pp. 76–91, Crete, Greece, May 10-12 2004. Springer.
9. Niles, I., and Pease A., Towards a standard upper ontology, 2001.
10. Levenshtein, V. I., Binary codes capable of correcting deletions, insertions, and reversals, 1966.
11. Wordnet: An electronic lexical database, 1998.
12. Calvanese, D., Giacomo, G. D., and Lenzerini, M., A frameworkfor ontology integration, 2001.
13. Zhdanova, A. V., and Shvaiko, P., Community-driven ontology matching. In 3rd European Semantic Web Conference, ESWC, pp. 34–49, Budva, Montenegro, June 11-14 2006.
14. http://www.w3.org/TR/rdf-sparql query.
15. Sean Bechhofer, Raphael Voltz, and Phillip Lord. Cooking the semantic web with the OWL API. In Proc. 2nd International Semantic Web Conference (ISWC), Sanibel Island (FL US), 2003.
16. Ian Horrocks, Peter Patel-Schneider, Harold Boley, Said Tabet, Benjamin Grosof, and Mike Dean. SWRL: a semantic web rule language combining OWL and RuleML, 2003. www.daml.org/2003/11/swrl/.
17. James Clark (editor),XSL Transformations (XSLT), Recommendation, W3C, 1999, http://www.w3.org/TR/xslt.
18. Jérôme Euzenat, Heiner Stuckenschmidt, and Mikalai Yatskevich. Introduction to the ontology alignment evaluation 2005. In K-CAP 2005 Integrating Ontologies Workshop, Banff, Alberta, Canada, 2005.
19. Do, H., Melnik, S., & Rahm, E., Comparison of schema matching evaluations. In Proceedings of the 2nd International Workshop on Web Databases (German Informatics Society), 2002.

Optimality and duality in nondifferentiable multiobjective programming under d_I-invexity

Hachem Slimani[1] and Mohammed Said Radjef[2]

[1] Laboratory of Modeling and Optimization of Systems LAMOS
University of Bejaia 06000, Algeria,
haslimani@yahoo.fr,
[2] Laboratory of Modeling and Optimization of Systems LAMOS
University of Bejaia 06000, Algeria
radjefms@yahoo.fr

Abstract. In this paper, we are concerned with a nondifferentiable multiobjective programming problem with inequality constraints. We introduce new concepts of d_I-invexity and generalized d_I-invexity in which each component of the objective and constraint functions is directionally differentiable in its own direction d_i. New Fritz-John type necessary and Karush-Kuhn-Tucker type necessary and sufficient optimality conditions are obtained for a feasible point to be weakly efficient, efficient or properly efficient. Moreover, we prove weak and strong duality results for a Mond-Weir type dual under various types of generalized d_I-invexity assumptions.
Keywords: Nondifferentiable multiobjective programming; Optimality; Duality; Directional derivative; d_I-invexity; d_I-V-type I; Weakly efficient and Efficient solution.

1 Introduction

In optimization theory convexity plays a very important role in many aspects of mathematical programming including sufficient optimality conditions and duality theorems see, for example, Mangasarian [13] and Bazaraa et al. [3]. Several classes of functions have been defined for the purpose of weakening the hypothesis of convexity in Mathematical programming. Hanson [8] introduced the concept of invexity for the differentiable functions, generalizing the difference $(x - x_0)$ in the definition of convex function to any function $\eta(x, x_0)$. Hanson and Mond [9] introduced two new classes of functions called Type I and Type II functions for the scalar optimization problem, which were further generalized to pseudo-Type I and quasi-Type I by Rueda and Hanson [19] and sufficient optimality conditions are obtained involving these functions.

On the other hand, Kaul et al. [12] considered a multiobjective problem involving generalized type-I functions, with scalarisation, and obtained some results on optimality and duality, where the Wolfe and Mond-Weir duals are considered. Aghezzaf and Hachimi [1] introduced new classes of generalized type-I vector-valued functions and, without scalarisation, derived various duality results

for a nonlinear multiobjective programming problem. Following Jeyakumar and Mond [11] and Kaul et al. [12], Hanson et al. [10] introduced the V-type I problem with respect to η, including positive real-valued functions α_i and β_j in their definition, and they obtained optimality conditions and duality results under various types of generalized V-type I requirements.

Jeyakumar and Mond [11] have observed that one major difficulty in all of these extensions of convexity is that invex problems require the same function $\eta(x, x_0)$ for the objective and constraint functions. This requirement turns out to be a major restriction in applications. In [17, 21], Radjef and Slimani are considered the invexity and its extensions for the objective and constraint functions with respect to different η_i (each function occurring in the problem is considered with respect to its own function η_i). Optimality conditions, for nonlinear and multiobjective programming problems, are obtained under this generalized invexity and a method how to find these functions η_i is established. In [20], for multiobjective programming, some duality results are obtained for a dual in the format of Mond-Weir under generalized V-type I invexity with respect to different $(\eta_i)_i$ and $(\theta_j)_j$. Further, in [17, 20, 21], a generalized Karush-Kuhn-Tucker relation is introduced which is necessary and sufficient for a feasible point to be optimal under various types of generalized invexity requirements. Using this generalized KKT-relation, in [17, 20, 21] numerical examples are constructed and show that the results obtained are applicable to prove that a feasible point is optimal even if this point is not Karush-Kuhn-Tucker stationary (resp. a vector Karush-Kuhn-Tucker) for a nonlinear (resp. multiobjective) programming problem.

For nondifferentiable programming, the invexity was generalized to locally Lipschitz functions with derivative replaced by the Clarke generalized gradient [5], see Craven [6], Reiland [18], Ye [25], Suneja and Srivastava [22], Antczak [2], Mishra et al. [14] and Mishra and Noor [15]. However, like in differentiable case, these extension of invexity is that d-invex problems require the same direction d for the directional derivatives of objective and constraint functions.

In the present paper, we define a new classe of nondifferentiable functions called d_I-invexity in which each component of the objective function is directionally differentiable in its own direction d_i. In the following, from Suneja and Srivastava [22], Mishra and Noor [15] and Slimani and Radjef [20], we introduce new classes of problems called d_I-V-type I, quasi-, pseudo-, pseudo quasi-, quasi pseudo- d_I-V-type I with respect to $(\eta_i)_i$ and $(\theta_j)_j$. New Fritz-John type necessary and Karush-Kuhn-Tucker type necessary and sufficient optimality conditions are obtained for a feasible point to be weakly efficient or efficient solution. Moreover, a Mond-Weir type dual is considered and weak and strong duality results are proved under various types of generalized d_I-invexity assumptions.

2 Preliminaries and definitions

The following conventions for equalities and inequalities will be used. If x, $y \in \mathbb{R}^n$, then

$x = y \Leftrightarrow x_i = y_i$, i=1,...,n;
$x < y \Leftrightarrow x_i < y_i$, i=1,...,n;
$x \leqq y \Leftrightarrow x_i \leqq y_i$, i=1,...,n;
$x \leq y \Leftrightarrow x \leqq y$ and $x \neq y$.
We also note $\mathbb{R}^q_{\leqq}$ (resp. $\mathbb{R}^q_{\geq}$ or $\mathbb{R}^q_{>}$) the set of vectors $y \in \mathbb{R}^q$ with $y \geqq 0$ (resp. $y \geq 0$ or $y > 0$).

Definition 1. *[4] Let $f : D \to \mathbb{R}$ be defined on a non-empty set $D \subset \mathbb{R}^n$. The function f is pre-invex (with respect to η) on D, if there exists a vector function $\eta : D \times D \to \mathbb{R}^n$ such that the inequality*

$$f(x_0 + \lambda\eta(x, x_0)) \leqq \lambda f(x) + (1 - \lambda)f(x_0), \tag{1}$$

holds for all $x, x_0 \in D$ and any $\lambda \in [0, 1]$.

Definition 2. *[2] A m-dimensional vector valued function $\phi : D \to \mathbb{R}^m$ is pre-invex with respect to η, if each of its components is pre-invex on D with respect to the same function η.*

Definition 3. *[5] The function $f : D \to \mathbb{R}$ is said to be locally Lipschitz at $x_0 \in D$, if there exists a constant $K > 0$ such that*

$$|f(y) - f(x)| \leqq K\|y - x\|, \ \forall\ x, y \text{ within a neighbourhood of } x_0,$$

where $\|.\|$ denotes the Euclidean norm. We say that f is locally Lipschitz on X if it is locally Lipschitz at any point of X.

Definition 4. *[5] If $f : D \to \mathbb{R}$ is locally Lipschitz at $x_0 \in D$, the generalized directional derivative (in the sense of Clarke) of f at x_0 in the direction $d \in \mathbb{R}^n$, denoted by $f^0(x_0; d)$, is given by*

$$f^0(x_0; d) = \lim_{y \to x_0} \sup_{t \to 0^+} \frac{f(y+td)-f(y)}{t}.$$

And the usual one-sided directional derivative of f at x_0 is defined by

$$f'(x_0; d) = \lim_{\lambda \to 0^+} \frac{f(x_0+\lambda d)-f(x_0)}{\lambda},$$

whenever this limit exists. Obviously, $f^0(x_0; d) \geqq f'(x_0; d)$.
f is directionally differentiable at x_0 if its directional derivative exists finite in all directions.

Definition 5. *[25] Let $f : D \to \mathbb{R}^N$ be defined on a non-empty open set $D \subset \mathbb{R}^n$ and directionally differentiable at $x_0 \in D$. f is called d-invex at x_0 on D with respect to η, if there exists a vector function $\eta : D \times D \to \mathbb{R}^n$, such that for any $x \in D$ and for $i = 1, ..., N$,*

$$f_i(x) - f_i(x_0) \geqq f_i'(x_0; \eta(x, x_0)), \tag{2}$$

where $f_i'(x_0; \eta(x, x_0))$ denotes the directional derivative of f_i in the direction $\eta(x, x_0)$: $f_i'(x_0; \eta(x, x_0)) = \lim_{\lambda \to 0^+} \frac{f_i(x_0+\lambda\eta(x,x_0))-f_i(x_0)}{\lambda}$.
if inequalities (2) are satisfied at any point $x_0 \in D$, then f is said to be d-invex on D with respect to η.

Ye [25] has defined the invexity in the nondifferentiable case for the function f by taking all its components directionally differentiable in the same direction d. In the following definition, we define the d_I-invexity at a point x_0 for f, where each component $f_i,\ i = 1, ..., N$ is directionally differentiable in its own direction d_i.

Definition 6. *Let $f : D \to \mathbb{R}^N$ be defined on a non-empty open set $D \subset \mathbb{R}^n$ and for all $i = 1, ..., N,\ f_i$ is directionally differentiable at $x_0 \in D$ (not necessary in all directions). f is called d_I-invex at x_0 on D with respect to $(\eta_i)_{i=\overline{1,N}}$, if there exists N vector functions $\eta_i : D \times D \to \mathbb{R}^n,\ i = 1, ..., N$, such that for any $x \in D$*

$$f_i(x) - f_i(x_0) \geqq f_i'(x_0; \eta_i(x, x_0)),\ \textit{for all } i = 1, ..., N, \tag{3}$$

where $f_i'(x_0; \eta_i(x, x_0))$ denotes the directional derivative of f_i in the direction $\eta_i(x, x_0)$: $f_i'(x_0; \eta_i(x, x_0)) = \lim\limits_{\lambda \to 0^+} \frac{f_i(x_0 + \lambda \eta_i(x, x_0)) - f_i(x_0)}{\lambda}$.
if inequalities (3) are satisfied at any point $x_0 \in D$, then f is said to be d_I-invex on D with respect to $(\eta_i)_{i=\overline{1,N}}$.

Remark 1. In the above definition, it doesn't require that the directional derivative exists finite in all directions for each component $f_i,\ i = 1, ..., N$. Further, it is not necessary that all the components to be directionally differentiable in the same direction, which may widen the area of application.

In order to illustrate the definition 6, we give the following example.

Consider the following multiobjective programming problem

$$\begin{array}{ll} (VP) & \text{Minimize } f(x) = (f_1(x), ..., f_N(x)), \\ & \text{subject to } \ g(x) \leqq 0, \end{array}$$

where $f : D \to \mathbb{R}^N,\ g : D \to \mathbb{R}^k$, D is a nonempty open subset of $\mathbb{R}^n$.
Let $X = \{x \in D / g(x) \leqq 0\}$, the set of feasible solutions of (VP) and for $x_0 \in X$, we denote $J(x_0) = \{j \in \{1, ..., k\} / g_j(x_0) = 0\}$, $\tilde{J}(x_0) = \{j \in \{1, ..., k\} / g_j(x_0) < 0\}$; It is obvious that $J(x_0) \cup \tilde{J}(x_0) = \{1, ..., k\}$.

Now following Suneja and Srivastava [22], Mishra and Noor [15], Slimani and Radjef [20] and the definition 6, we define the generalized d-V-type I problems.

Definition 7. *We say that the problem (VP) is of d_I-V-type I at $x_0 \in X$ with respect to $(\eta_i)_{i=\overline{1,N}}$ and $(\theta_j)_{j=\overline{1,k}}$, if there exists $(N + k)$ vector functions $\eta_i : X \times X \to \mathbb{R}^n,\ i = \overline{1, N}$ and $\theta_j : X \times X \to \mathbb{R}^n,\ j = \overline{1, k}$ such that for all $x \in X$:*

$$f_i(x) - f_i(x_0) \geqq f_i'(x_0; \eta_i(x, x_0)),\ \forall\ i = 1, ..., N, \tag{4}$$

$$-g_j(x_0) \geqq g_j'(x_0; \theta_j(x, x_0)),\ \forall\ j = 1, ..., k. \tag{5}$$

If strict inequality hold in (4)(whenever $x \neq x_0$), we say that (VP) is of semi strictly d_I-V-type I at x_0 with respect to $(\eta_i)_{i=\overline{1,N}}$ and $(\theta_j)_{j=\overline{1,k}}$.

Definition 8. *We say that the problem (VP) is of quasi d_I-V-type I at $x_0 \in X$ with respect to $(\eta_i)_{i=\overline{1,N}}$ and $(\theta_j)_{j=\overline{1,k}}$, if there exists $(N+k)$ vector functions $\eta_i : X \times X \to \mathbb{R}^n,\ i = \overline{1,N}$ and $\theta_j : X \times X \to \mathbb{R}^n,\ j = \overline{1,k}$ such that for some vectors $\mu \in \mathbb{R}^N_{\geqq}$ and $\lambda \in \mathbb{R}^k_{\geqq}$:*

$$\sum_{i=1}^{N} \mu_i [f_i(x) - f_i(x_0)] \leqq 0 \quad \Rightarrow \quad \sum_{i=1}^{N} \mu_i f_i'(x_0; \eta_i(x, x_0)) \leqq 0,\ \forall\ x \in X, \tag{6}$$

$$\sum_{j=1}^{k} \lambda_j g_j(x_0) \geqq 0 \quad \Rightarrow \quad \sum_{j=1}^{k} \lambda_j g_j'(x_0; \theta_j(x, x_0)) \leqq 0,\ \forall\ x \in X. \tag{7}$$

If the second (implied) inequality in (6) is strict ($x \neq x_0$), we say that (VP) is of semi strictly quasi d_I-V-type I at x_0 with respect to $(\eta_i)_{i=\overline{1,N}}$ and $(\theta_j)_{j=\overline{1,k}}$.

Definition 9. *We say that the problem (VP) is of pseudo d_I-V-type I at $x_0 \in X$ with respect to $(\eta_i)_{i=\overline{1,N}}$ and $(\theta_j)_{j=\overline{1,k}}$, if there exists $(N+k)$ vector functions $\eta_i : X \times X \to \mathbb{R}^n,\ i = \overline{1,N}$ and $\theta_j : X \times X \to \mathbb{R}^n,\ j = \overline{1,k}$ such that for some vectors $\mu \in \mathbb{R}^N_{\geqq}$ and $\lambda \in \mathbb{R}^k_{\geqq}$:*

$$\sum_{i=1}^{N} \mu_i f_i'(x_0; \eta_i(x, x_0)) \geqq 0 \quad \Rightarrow \quad \sum_{i=1}^{N} \mu_i [f_i(x) - f_i(x_0)] \geqq 0,\ \forall\ x \in X, \tag{8}$$

$$\sum_{j=1}^{k} \lambda_j g_j'(x_0; \theta_j(x, x_0)) \geqq 0 \quad \Rightarrow \quad \sum_{j=1}^{k} \lambda_j g_j(x_0) \leqq 0,\ \forall\ x \in X. \tag{9}$$

If the second (implied) inequality in (8) (resp. (9)) is strict ($x \neq x_0$), we say that (VP) is of semi strictly pseudo d_I-V-type I in f (resp. g) at x_0 with respect to $(\eta_i)_{i=\overline{1,N}}$ and $(\theta_j)_{j=\overline{1,k}}$. If the second (implied) inequalities in (8) and (9) are both strict, we say that (VP) is of strictly pseudo d_I-V-type I at x_0 with respect to $(\eta_i)_{i=\overline{1,N}}$ and $(\theta_j)_{j=\overline{1,k}}$.

Definition 10. *We say that the problem (VP) is of quasi pseudo d_I-V-type I at $x_0 \in X$ with respect to $(\eta_i)_{i=\overline{1,N}}$ and $(\theta_j)_{j=\overline{1,k}}$, if there exists $(N+k)$ vector functions $\eta_i : X \times X \to \mathbb{R}^n, i = \overline{1,N}$ and $\theta_j : X \times X \to \mathbb{R}^n,\ j = \overline{1,k}$ such that for some vectors $\mu \in \mathbb{R}^N_{\geqq}$ and $\lambda \in \mathbb{R}^k_{\geqq}$:*

$$\sum_{i=1}^{N} \mu_i [f_i(x) - f_i(x_0)] \leqq 0 \quad \Rightarrow \quad \sum_{i=1}^{N} \mu_i f_i'(x_0; \eta_i(x, x_0)) \leqq 0,\ \forall\ x \in X, \tag{10}$$

$$\sum_{j=1}^{k} \lambda_j g_j'(x_0; \theta_j(x, x_0)) \geqq 0 \quad \Rightarrow \quad \sum_{j=1}^{k} \lambda_j g_j(x_0) \leqq 0,\ \forall\ x \in X. \tag{11}$$

If the second (implied) inequality in (11) is strict ($x \neq x_0$), we say that (VP) is of quasi strictly-pseudo d_I-V-type I at x_0 with respect to $(\eta_i)_{i=\overline{1,N}}$ and $(\theta_j)_{j=\overline{1,k}}$.

Definition 11. *We say that the problem (VP) is of pseudo quasi d_I-V-type I at $x_0 \in X$ with respect to $(\eta_i)_{i=\overline{1,N}}$ and $(\theta_j)_{j=\overline{1,k}}$, if there exists $(N+k)$ vector functions $\eta_i : X \times X \to \mathbb{R}^n, i = \overline{1,N}$ and $\theta_j : X \times X \to \mathbb{R}^n,\ j = \overline{1,k}$ such that for some vectors $\mu \in \mathbb{R}^N_{\geqq}$ and $\lambda \in \mathbb{R}^k_{\geqq}$:*

$$\sum_{i=1}^{N} \mu_i f_i'(x_0; \eta_i(x, x_0)) \geqq 0 \quad \Rightarrow \quad \sum_{i=1}^{N} \mu_i [f_i(x) - f_i(x_0)] \geqq 0,\ \forall\ x \in X, \tag{12}$$

$$\sum_{j=1}^{k} \lambda_j g_j(x_0) \geqq 0 \quad \Rightarrow \quad \sum_{j=1}^{k} \lambda_j g_j'(x_0; \theta_j(x, x_0)) \leqq 0,\ \forall\ x \in X. \tag{13}$$

If the second (implied) inequality in (12) is strict ($x \neq x_0$), we say that (VP) is of strictly-pseudo quasi d_I-V-type I at x_0 with respect to $(\eta_i)_{i=\overline{1,N}}$ and $(\theta_j)_{j=\overline{1,k}}$.

Remark 2. The problems defined above are different from those in Suneja and Srivastava [22], Mishra et al. [14], Mishra and Noor [15]. For example of differentiable generalized V-type I problems, one can refer to Slimani and Radjef [20].

3 Optimality conditions

In recent paper, Slimani and Radjef [20] considered a number of necessary and sufficient optimality conditions which depend on generalized V-type I problems with respect to $(\eta_i)_{i=\overline{1,N}}$ and $(\theta_j)_{j=\overline{1,k}}$. In this section, we adapt their results to the classes of nondifferentiable problems.

Theorem 1. *(Sufficiency) Let x_0 be a feasible solution for (VP) and suppose that there exists $(N+J)$ vector functions $\eta_i : X \times X \to \mathbb{R}^n,\ i = \overline{1,N}$, $\theta_j : X \times X \to \mathbb{R}^n,\ j \in J(x_0)$ and scalars $\mu_i \geqq 0,\ i = \overline{1,N},\ \sum_{i=1}^{N} \mu_i = 1$, $\lambda_j \geqq 0,\ j \in J(x_0)$ such that the following relation is satisfied:*

$$\sum_{i=1}^{N} \mu_i f_i'(x_0; \eta_i(x, x_0)) + \sum_{j \in J(x_0)} \lambda_j g_j'(x_0; \theta_j(x, x_0)) \geqq 0,\ \forall\ x \in X. \tag{14}$$

Moreover, assume that one of the following conditions is satisfied:

(a) the problem (VP) is quasi strictly-pseudo d_I-V-type I at x_0 with respect to $(\eta_i)_{i=\overline{1,N}}$ and $(\theta_j)_{j \in J(x_0)}$;
(b) the problem (VP) is semi strictly-quasi d_I-V-type I at x_0 with respect to $(\eta_i)_{i=\overline{1,N}}$, $(\theta_j)_{j \in J(x_0)}$ and for μ and λ;
(c) the problem (VP) is strictly-pseudo d_I-V-type I at x_0 with respect to $(\eta_i)_{i=\overline{1,N}}$, $(\theta_j)_{j \in J(x_0)}$ and for μ and λ.

Then x_0 is an efficient solution for (VP).

Proof. Suppose that x_0 is not an efficient solution of (VP). Then there exists a feasible solution $x \in X$ of (VP) such that $f(x) \leq f(x_0)$, which implies that

$$\sum_{i=1}^{N} \mu_i [f_i(x) - f_i(x_0)] \leqq 0. \tag{15}$$

From the above inequality and the condition (a), we obtain

$$\sum_{i=1}^{N} \mu_i f_i'(x_0; \eta_i(x, x_0)) \leqq 0. \tag{16}$$

By using the relation (14), we deduce that

$$\sum_{j \in J(x_0)} \lambda_j g_j'(x_0; \theta_j(x, x_0)) \geqq 0, \tag{17}$$

which implies, from the condition (a) (in view of definition 10), that

$$\sum_{j \in J(x_0)} \lambda_j g_j(x_0) < 0.$$

The last inequality contradicts the fact that $g_j(x_0) = 0,\ \forall\ j \in J(x_0)$ and hence the conclusion follows.
The proof of the part (b) is very similar to the proof of part (a), except that for this case the inequality (16) becomes strict ($<$), it follows that the inequality (17) becomes strict ($>$) and, using the reverse implication in (7), we get the contradiction again.
By condition (c), from $g_j(x_0) = 0,\ \lambda_j \geqq 0,\ \forall\ j \in J(x_0)$, in view of the reverse implication in (9), we obtain

$$\sum_{j \in J(x_0)} \lambda_j g_j'(x_0; \theta_j(x, x_0)) < 0,\ \forall\ x \in X \setminus \{x_0\}.$$

By using the relation (14), we deduce

$$\sum_{i=1}^{N} \mu_i f_i'(x_0; \eta_i(x, x_0)) > 0,\ \forall\ x \in X \setminus \{x_0\},$$

which implies, according to the relation (8) (for strictly-pseudo d_I-V-type I problem), that

$$\sum_{i=1}^{N} \mu_i [f_i(x) - f_i(x_0)] > 0,\ \forall\ x \in X \setminus \{x_0\}. \tag{18}$$

Thus (15) and (18) contradict each other, hence x_0 is an efficient solution of (VP). This completes the proof.

To prove the necessary condition, we need to prove the following lemma.

Lemma 1. *Suppose that*

(i) x_0 *is a (local) weakly efficient solution for (VP);*
(ii) g_j *is continuous at* x_0 *for* $j \in \tilde{J}(x_0)$ *and there exists vector functions* $\eta_i : X \times X \to \mathbb{R}^n,\ i = \overline{1,N}$, $\theta_j : X \times X \to \mathbb{R}^n,\ j \in J(x_0)$ *which satisfy at* x_0 *with respect to* $\eta : X \times X \to \mathbb{R}^n$ *the following inequalities for all* $x \in X$,

$$f_i'(x_0; \eta(x, x_0)) \leqq f_i'(x_0; \eta_i(x, x_0)),\ i = \overline{1,N}, \tag{19}$$

$$g_j'(x_0; \eta(x, x_0)) \leqq g_j'(x_0; \theta_j(x, x_0)),\ j \in J(x_0). \tag{20}$$

Then the system

$$f_i'(x_0; \eta_i(x, x_0)) < 0,\ i = \overline{1,N}, \tag{21}$$

$$g_j'(x_0; \theta_j(x, x_0)) < 0,\ j \in J(x_0), \tag{22}$$

has no solution $x \in X$.

Proof. Let x_0 be a local weakly efficient solution for (VP) and suppose there exists $\tilde{x} \in X$ such that the inequalities (21) and (22) are true.
For $i \in \{1, ..., N\}$, let $\varphi_{f_i}(x_0, \tilde{x}, \tau) = f_i(x_0 + \tau\eta(\tilde{x}, x_0)) - f_i(x_0)$.
We observe that this function vanishes at $\tau = 0$ and
$\lim\limits_{\tau \to 0} \frac{\varphi_{f_i}(x_0, \tilde{x}, \tau) - \varphi_{f_i}(x_0, \tilde{x}, 0)}{\tau} = \lim\limits_{\tau \to 0} \frac{f_i(x_0 + \tau\eta(\tilde{x}, x_0)) - f_i(x_0)}{\tau} = f_i'(x_0; \eta(\tilde{x}, x_0)) \leqq f_i'(x_0; \eta_i(\tilde{x}, x_0)) < 0$ using (19) and (21).
It follows that, for all $i \in \{1, ..., N\},\ \varphi_{f_i}(x_0, \tilde{x}, \tau) < 0$ if τ is in some open interval $(0, \delta_{f_i})$. Thus, for all $i \in \{1, ..., N\}$,

$$f_i(x_0 + \tau\eta(\tilde{x}, x_0)) < f_i(x_0),\ \tau \in (0, \delta_{f_i}).$$

Similarly, by defining $\varphi_{g_j}(x_0, \tilde{x}, \tau) = g_j(x_0 + \tau\eta(\tilde{x}, x_0)) - g_j(x_0),\ j \in J(x_0)$ and using (20) and (22), we can prove that $g_j(x_0 + \tau\eta(\tilde{x}, x_0)) < g_j(x_0),\ j \in J(x_0),\ \tau \in (0, \delta_{g_j})$ and by definition of $J(x_0)$, we obtain

$$g_j(x_0 + \tau\eta(\tilde{x}, x_0)) < 0,\ j \in J(x_0),\ \tau \in (0, \delta_{g_j}).$$

Since g_j is continuous at x_0 for $j \in \tilde{J}(x_0)$, therefore, there exists δ_j such that

$$g_j(x_0 + \tau\eta(\tilde{x}, x_0)) < 0,\ j \in \tilde{J}(x_0),\ \tau \in (0, \delta_j).$$

Let $\delta_0 = \min\{\delta_{f_i},\ i = \overline{1,N},\ \delta_{g_j},\ j \in J(x_0),\ \delta_j,\ j \in \tilde{J}(x_0)\}$.
Then $(x_0 + \tau\eta(\tilde{x}, x_0)) \in N_{\delta_0}(x_0),\ \tau \in (0, \delta_0)$, where $N_{\delta_0}(x_0)$ is a neighborhood of x_0. Now

$$f_i(x_0 + \tau\eta(\tilde{x}, x_0)) < f_i(x_0),\ i = \overline{1,N}, \tag{23}$$

$$g_j(x_0 + \tau\eta(\tilde{x}, x_0)) < 0,\ j = \overline{1,k}. \tag{24}$$

By (23) and (24), we get $(x_0 + \tau\eta(\tilde{x}, x_0)) \in N_{\delta_0}(x_0) \cap X,\ \tau \in (0, \delta_0)$.
Hence (23) is a contradiction to the assumption that x_0 is a (local) weakly efficient solution for (VP). Thus there exists no $x \in X$ satisfying the system (21) and (22).

The following lemma given by Weir and Mond [23] will be used.

Lemma 2. *Let S be a nonempty set in $\mathbb{R}^n$ and let $\psi : S \to \mathbb{R}^m$ be a pre-invex function on S (with respect to η). Then either*

$\psi(x) < 0$ *has a solution* $x \in S$,

or

$p^t\psi(x) \geqq 0$ *for all* $x \in S$, *for some* $p \in \mathbb{R}^m_{\geq}$,

but both alternatives are never true.

Antczak [2] has given necessary conditions for $x_0 \in X$ to be weakly efficient solution for (VP) by taking the functions $f_i,\ i = \overline{1,N}$ and $g_j,\ j \in J(x_0)$ directionally differentiable in the same direction $\eta(x, x_0)$. Now we give necessary optimality criteria by considering each function $f_i,\ i = \overline{1,N}$ (resp. $g_j,\ j \in J(x_0)$) directionally differentiable in its own direction $\eta_i(x, x_0),\ i = \overline{1,N}$ (resp. $\theta_j(x, x_0),\ j \in J(x_0)$).

Theorem 2. *(Fritz John type necessary optimality conditions) Suppose that*

(i) x_0 *is a weakly efficient solution for (VP);*

(ii) g_j *is continuous at* x_0 *for* $j \in J(x_0)$ *and there exists vector functions* $\eta_i : X \times X \to \mathbb{R}^n,\ i = \overline{1,N},\ \theta_j : X \times X \to \mathbb{R}^n,\ j \in J(x_0)$ *which satisfy at* x_0 *with respect to* $\eta : X \times X \to \mathbb{R}^n$ *the following inequalities for all* $x \in X$,

$$f_i'(x_0; \eta(x, x_0)) \leqq f_i'(x_0; \eta_i(x, x_0)),\ i = \overline{1,N}, \tag{25}$$

$$g_j'(x_0; \eta(x, x_0)) \leqq g_j'(x_0; \theta_j(x, x_0)),\ j \in J(x_0); \tag{26}$$

(iii) the functions $f_i'(x_0; \eta_i(x, x_0)),\ i = \overline{1,N}$ *and* $g_j'(x_0; \theta_j(x, x_0)),\ j \in J(x_0)$ *are pre-invex of* x *on* X *with respect to* $\gamma : X \times X \to \mathbb{R}^n$;

Then there exists $(\mu, \lambda) \in \mathbb{R}^{N+|J(x_0)|}_{\geq}$ *such that* (x_0, μ, λ) *satisfies*

$$\sum_{i=1}^{N} \mu_i f_i'(x_0; \eta_i(x, x_0)) + \sum_{j \in J(x_0)} \lambda_j g_j'(x_0; \theta_j(x, x_0)) \geqq 0,\ \forall\ x \in X. \tag{27}$$

Proof. If the conditions (i) and (ii) are satisfied, then, by lemma 1 the system (21)-(22) has no solution for $x \in X$.
Since, by hypothesis (iii), $f_i'(x_0; \eta_i(x, x_0)),\ i = \overline{1,N}$ and $g_j'(x_0; \theta_j(x, x_0)),\ j \in J(x_0)$ are pre-invex of x on X with respect to γ, therefore, by lemma 2, there exists $p = (\mu, \lambda) \in \mathbb{R}^{N+|J(x_0)|}_{\geq}$ such that the relation (27) is satisfied.

Now, we need a constraint qualification to prove the next result.

Definition 12. *The nondifferentiable function g is said to satisfy the d_I-constraint qualification at $x_0 \in X$ with respect to $(\theta_j)_{j \in J(x_0)}$ if there exists $\bar{x} \in X$ and $\theta_j : X \times X \to \mathbb{R}^n,\ j \in J(x_0)$ such that*

$$g_j'(x_0; \theta_j(\bar{x}, x_0)) < 0,\ \forall\ j \in J(x_0). \tag{28}$$

Note that in the d_I-constraint qualification, we don't require that $g_j(\bar{x}) < 0,\ \forall\ j \in J(x_0)$ as in [2, 15].

Thus, the following Karush-Kuhn-Tucker type necessary optimality conditions for (VP) are satisfied.

Theorem 3. *(Karush-Kuhn-Tucker type necessary optimality conditions) Suppose that the conditions (i), (ii) and (iii) of Theorem 2 are satisfied and the function g satisfies the d_I-constraint qualification at $x_0 \in X$ with respect to $(\theta_j)_{j\in J(x_0)}$. Then there exists $\mu \in \mathbb{R}^N_{\geq}$ and $\lambda \in \mathbb{R}^{|J(x_0)|}_{\geqq}$ such that (x_0, μ, λ) satisfies the relation (27).*

Proof. Based on Theorem 2 we obtain that there exists $(\mu, \lambda) \in \mathbb{R}^{N+|J(x_0)|}_{\geq}$ such that (x_0, μ, λ) satisfies the relation (27). Now it is enough to prove that $\mu \neq 0$. We proceed by contradiction. If $\mu = 0$, then $\lambda \neq 0$ and (27) takes the following form

$$\sum_{j\in J(x_0)} \lambda_j g'_j(x_0; \theta_j(x, x_0)) \geqq 0,\ \forall\ x \in X,$$

which contradicts the d_I-constraint qualification. Hence $\mu \neq 0$.

4 Mond-Weir duality

In relation to (VP) and using the relation (27), we consider the following multiobjective dual problem, which is in the format of Mond-Weir [16].

$$(MWD) \qquad \text{Maximize } f(y) = (f_1(y), ..., f_N(y)),$$

subject to

$$\sum_{i=1}^{N} \mu_i f'_i(y; \eta_i(x, y) + \sum_{j\in J(y)} \lambda_j g'_j(y; \theta_j(x, y)) \geqq 0,\ \forall\ x \in X. \tag{29}$$

$$y \in X,\ \ \mu \in\ \mathbb{R}^N_{\geq},\ \ \lambda \in \mathbb{R}^{|J(y)|}_{\geqq},$$
$$\eta_i : X \times X \to \mathbb{R}^n,\ \forall\ i = 1, ..., N,$$
$$\theta_j : X \times X \to \mathbb{R}^n,\ \forall\ j \in J(y).$$

We let Y be the set of feasible solutions of problem (MWD); ie,

$$Y = \Bigg\{ (y, \mu, \lambda, (\eta_i)_i, (\theta_j)_j) : \sum_{i=1}^{N} \mu_i f'_i(y; \eta_i(x, y)) + \sum_{j\in J(y)} \lambda_j g'_j(y; \theta_j(x, y)) \geqq 0,$$
$$\forall\ x \in X;\ y \in X,\ \mu \in\ \mathbb{R}^N_{\geq},\ \lambda \in \mathbb{R}^{|J(y)|}_{\geqq};\ \eta_i : X \times X \to \mathbb{R}^n,\ \forall\ i = 1, ..., N;$$
$$\theta_j : X \times X \to \mathbb{R}^n,\ \forall\ j \in J(y)\}.$$

We denote by $Pr_X Y$ the projection of set Y on X.

Theorem 4. *(Weak duality) Let x_0 and $(y, \mu, \lambda, (\eta_i)_{i=\overline{1,N}}, (\theta_j)_{j\in J(y)})$ be feasible solutions for (VP) and (MWD) respectively. Moreover, assume that one of the following conditions is satisfied:*

(a) the problem (VP) is pseudo quasi d_I-V-type I at y with respect to $(\eta_i)_{i=\overline{1,N}}$, $(\theta_j)_{j\in J(y)}$ and for μ and λ (with $\mu > 0$);
(b) the problem (VP) is strictly-pseudo quasi d_I-V-type I at y with respect to $(\eta_i)_{i=\overline{1,N}}$, $(\theta_j)_{j\in J(y)}$ and for μ and λ.
(c) the problem (VP) is quasi strictly-pseudo d_I-V-type I at y with respect to $(\eta_i)_{i=\overline{1,N}}$, $(\theta_j)_{j\in J(y)}$ and for μ and λ.

Then $f(x) \nleqq f(y)$.

Proof. By condition (a) (in view of definition 11), it follows that

$$\sum_{j\in J(y)} \lambda_j g_j'(y; \theta_j(x, y)) \leqq 0. \tag{30}$$

Since $(y, \mu, \lambda, (\eta_i)_{i=\overline{1,N}}, (\theta_j)_{j\in J(y)})$ is feasible for (MWD), (30) implies that

$$\sum_{i=1}^{N} \mu_i f_i'(y; \eta_i(x, y)) \geqq 0. \tag{31}$$

Condition (a) and inequality (31) give

$$\sum_{i=1}^{N} \mu_i [f_i(x) - f_i(y)] \geqq 0. \tag{32}$$

Now suppose to the contrary that $f(x) \leq f(y)$. Then, since $\mu > 0$, we obtain

$$\sum_{i=1}^{N} \mu_i [f_i(x) - f_i(y)] < 0, \tag{33}$$

which contradicts (32). Hence the conclusion follows.
The proof of the parts (b) and (c) are very similar to the proof of part (a), except that: For the part (b), the inequality (32) becomes strict ($>$) and the inequality (33) becomes non strict ($\leqq$). For the part (c), the inequality (30) becomes strict ($<$), it follows that the inequalities (31) and (32) become strict ($>$). Since $\mu \geqq 0$, then the inequality (33) becomes non strict ($\leqq$). In the two cases, the inequalities (32) and (33) contradict each other always. This completes the proof.

Remark 3. If we omit the assumption $\mu > 0$ in the condition (a) or the word strictly in the condition (b), we obtain, for this part of theorem, $f(x) \not< f(y)$.

Theorem 5. *(Strong duality) Let x_0 be weakly efficient solution for (VP) and suppose that the conditions (ii) and (iii) of Theorem 2 are satisfied. Assume*

also that the function g satisfies the d_I-constraint qualification at x_0 with respect to $(\theta_j)_{j\in J(x_0)}$. Then there exists $\mu \in \mathbb{R}^N_{\geq}$ and $\lambda \in \mathbb{R}^{|J(x_0)|}_{\geqq}$ such that $(x_0, \mu, \lambda, (\eta_i)_{i=\overline{1,N}}, (\theta_j)_{j\in J(x_0)}) \in Y$ and the objective functions of (VP) and (MWD) have the same values at x_0 and $(x_0, \mu, \lambda, (\eta_i)_{i=\overline{1,N}}, (\theta_j)_{j\in J(x_0)})$, respectively. If, further, the weak duality with the condition (a) without ($\mu > 0$) (resp. with the condition (b) or (c)) between (VP) and (MWD) in theorem 4 holds, then $(x_0, \mu, \lambda, (\eta_i)_{i=\overline{1,N}}, (\theta_j)_{j\in J(x_0)}) \in Y$ is a weakly efficient (resp. an efficient) solution of (MWD).

Proof. By the theorem 3, there exists $\mu \in \mathbb{R}^N_{\geq}$ and $\lambda \in \mathbb{R}^{|J(x_0)|}_{\geqq}$ such that

$$\sum_{i=1}^{N} \mu_i f_i'(x_0; \eta_i(x, x_0)) + \sum_{j\in J(x_0)} \lambda_j g_j'(x_0; \theta_j(x, x_0)) \geqq 0, \ \forall\ x \in X.$$

It follows that $(x_0, \mu, \lambda, (\eta_i)_{i=\overline{1,N}}, (\theta_j)_{j\in J(x_0)}) \in Y$.
Trivially, the objective function values of (VP) and (MWD) are equal.
Next suppose that $(x_0, \mu, \lambda, (\eta_i)_{i=\overline{1,N}}, (\theta_j)_{j\in J(x_0)}) \in Y$ is not a weakly efficient (resp. an efficient) solution of (MWD). Then there exists $(y^*, \mu^*, \lambda^*, (\eta_i^*)_{i=\overline{1,N}}, (\theta_j^*)_{j\in J(y^*)}) \in Y$ such that $f(x_0) < f(y^*)$ (resp. $f(x_0) \leq f(y^*)$), which violates the weak duality theorem 4 (see the remark 3).
Hence $(x_0, \mu, \lambda, (\eta_i)_{i=\overline{1,N}}, (\theta_j)_{j\in J(x_0)}) \in Y$ is indeed a weakly efficient (resp. an efficient) solution of (MWD).

5 Conclusion

In this paper, we have defined a new classe of nondifferentiable functions called d_I-invexity in which each component of the objective function is directionally differentiable in its own direction d_i. In the following, from Suneja and Srivastava [22], Mishra and Noor [15], Slimani and Radjef [20], we have introduced new classes of problems called d_I-V-type I, quasi-, pseudo-, pseudo quasi-, quasi pseudo- d_I-V-type I with respect to $(\eta_i)_i$ and $(\theta_j)_j$. In the framework of the new concepts, we prove new Fritz-John type necessary and Karush-Kuhn-Tucker type necessary and sufficient optimality conditions for a feasible point to be weakly efficient or efficient solution. Moreover, we prove weak and strong duality results for Mond-Weir type dual under various types of generalized d_I-V-type I requirements.

References

1. Aghezzaf, B., Hachimi, A.: Generalized Invexity and Duality in Multiobjective Programming Problems, J. of Global Optim. **18** (2000), 91-101.
2. Antczak, T.: Multiobjective Programming under d-invexity, European Journal of Operational Research **137** (2002), 28-36.
3. Bazaraa, M.S., Sherali, H.D., Shetty, C.M.: Nonlinear Programming: Theory and Algorithms, Wiley, New York, 1991.

4. Ben-Israel, A., Mond, B.: What is Invexity ?, J. Austral. Math. Soc. Ser. **B 28** (1986), 1-9.
5. Clarke, F.H., Optimization and Nonsmooth Analysis, Wiley, New York, 1983.
6. Craven, B.D.: Nondifferentiable optimization by smooth approximation, Optimization **17** (1986) 3–17.
7. Geoffrion, A.M., Proper Efficiency and the Theory of Vector Maximisation, J. Math. Anal. Appl. **22** (1968), 618-630.
8. Hanson, M.A.: On Sufficiency of the Kuhn-Tuker Conditions, J. Math. Anal. Appl. **80** (1981), 445-550.
9. Hanson, M.A., Mond, B.: Necessary and Sufficient Conditions in Constrained Optimisation, Mathematical programming **37** (1987), 51-58.
10. Hanson, M.A., Pini, R., Singh, C.: Multiobjective Programming Under Generalized Type I Invexity, J. Math. Anal. Appl. **261** (2001), 562-577.
11. Jeyakumar, V., Mond, B.: On generalized convex mathematical programming, J. Austral. Math. Soc. Ser. **B 34** (1992), 43–53.
12. Kaul, R.N., Suneja, S.K., Srivastava, M.K.: Optimality Criteria and Duality in Multiple-Objective Optimization Involving Generalized Invexity, J. Optim. Theo. Appl. **80** (1994), 465-482.
13. Mangasarian, O.L.: Nonlinear Programming, McGraw–Hill, New York, 1969.
14. Mishra, S.K., Wang, S.Y., Lai, K.K.: Optimality and Duality in Nondifferentiable and Multiobjective Programming under Generalized d-Invexity, J. Global Optim. **29** (2004), 425-438.
15. Mishra, S.K., M.A. Noor: Some nondifferentiable multiobjective programming problems, J. Math. Anal. Appl. **316** (2006), 472-482.
16. Mond, B., Weir, T.: Generalized Concavity and Duality, in Generalized Concavity in Optimization and Economics, edited by S. Schaible and W.T. Ziemba, Acadimic Press, New York, (1981), 263-276.
17. Radjef, M.S., Slimani, H.: Optimality in nonlinear programming under generalized Karush-Kuhn-Tucker relation, Submited for publication in J. Optim. Theory Appl. october 2006.
18. Reiland, T.W.: Nonsmooth invexity, Bull. Austral. Math. Soc. **42** (1990) 437–446.
19. Rueda, N.G., Hanson, M.A.: Optimality criteria in mathematical programming involving generalized invexity, J. Math. Anal. Appl. **130** (1988), 375-385.
20. Slimani, H., Radjef, M.S.: Generalized V-type I Invexity and Duality for Multiobjective programs, Presented at the 8th International Symposium on Generalized Convexity and Monotonicity, July 4-8, 2005 Varese, Italy. Submited for publication in J. Global Optim. March 2007.
21. Slimani, H., Radjef, M.S.: Multiobjective Programming under invexity with respect to different η_i, Presented at "Colloque International sur l'Optimisation et les Systèmes d'Information (COSI 2005)" June 11-14, 2005 University of Bejaia, Algeria.
22. Suneja, S.K., Srivastava, M.K.: Optimality and Duality in Nondifferentiable Multiobjective Optimization Involving d-Type I and Related Functions, J. Math. Anal. Appl. **206** (1997), 465-479.
23. Weir, T., Mond, B.: Pre-invex Functions in Multiple Objective Optimization, J. Math. Anal. Appl. **136** (1988), 29-38.
24. Wolfe, P.: A duality theorem for nonlinear programming, Quart. Appl. Math. **19** (1961), 239-244.
25. Ye, Y.L.: d-invexity and optimality conditions, J. Math. Anal. Appl. **162** (1991), 242-249.

OPTIMISATION PAR ESSAIM DE PARTICULES : APPLICATION AU PROBLEME DU VOYAGEUR DE COMMERCE

FAREH Abdelhak

Département d'Informatique
Université Mohamed Khider – Biskra – Algérie
hakou_alsahafi@yahoo.fr

Résumé. Nous présentons dans ce papier, l'une des méthodes de la large catégorie des méthodes d'intelligence en essaim qui sont utilisées pour résoudre les problèmes d'optimisation combinatoire. Cette méthode est appelée l'optimisation par essaim de particules. C'est une technique peu connue, fondée sur la coopération entre des agents (les particules), qui peuvent être vus comme des « animaux » aux capacités assez limitées (peu de mémoire et de facultés de raisonnement). L'échange d'informations entre ces agents rudimentaires permet de résoudre des problèmes difficiles, particulièrement les problèmes d'optimisation non linéaire, à variables continues, entières ou mixtes. Afin de mieux comprendre à la fois l'efficacité et les limites de cette approche, nous l'avons appliquée au problème du voyageur de commerce.

Mots clés. Optimisation par essaim de particules, métaheuristiques, vie artificielle, intelligence en essaim, voyageur de commerce.

Abstract. We present in this paper, one of the methods of the wide category of methods of swarm intelligence which are used to solve the problems of combinatorial optimization. This method is called particle swarm optimization. It is a technique little known, founded on the cooperation between agents (the particles), which will be seen as « animals » with an enough limited capacities (bit of memory and of ability of reasoning). The exchange of information between these rudimentary agents permits to solve the difficult problems, in particular the non linear optimization problems, with a continuous variables, entire or mixed. In order to better understand and in the same time the efficiency and the limits of this approach, we applied it to the traveling salesman problem.

Keywords. Particle swarm optimization, metaheuristics, artificial life, swarm intelligence, traveling salesman.

1. INTRODUCTION

Le comportement collectif des insectes sociaux reste une source d'inspiration pour les chercheurs. L'un des buts principaux de ces recherches sur les comportements collectifs des créatures naturelles est de trouver des méthodes puissantes pour la conception d'algorithmes d'optimisation combinatoire.

Par exemple, les fourmis ont collectivement la capacité de trouver le plus court chemin entre leur nid et une source de nourriture. L'étude de leur comportement a pu démontrer que la coopération au sein de la colonie est auto-organisée, résultant d'interaction entre les individus. Ces interactions, souvent très simples, permettent à la colonie de résoudre des problèmes compliqués. Ce phénomène, appelé intelligence en essaim [1], est un cadre très général et regroupe de nombreux algorithmes à base de population d'agents, dans lequel s'inscrivent les algorithmes de colonies de fourmis, d'optimisation par essaim de particules et autres.

La méthode d'optimisation par essaim de particules a d'ailleurs des similarités avec les algorithmes de colonies de fourmis. Cette idée veut qu'un groupe d'individus peu intelligents peut posséder une organisation globale complexe.

Ainsi, grâce à des règles de déplacement très simples (dans l'espace des solutions), les particules peuvent converger progressivement vers un minimum local. Cette méta-heuristique semble cependant mieux fonctionner pour des espaces en variables continues [2].

Dans ce papier, nous commençons par donner une vue d'ensemble sur la méthode d'optimisation par essaim de particules, son origine, son modèle de base, ses formulations, ses paramètres et son algorithme. Nous présentons ensuite le problème du voyageur de commerce suivi par une validation de l'approche d'optimisation par essaim de particules à ce problème.

2. L'OPTIMISATION PAR ESSAIM DE PARTICULES (OEP)

2.1. ORIGINE DE L'OEP

Les algorithmes (« d'Optimisation par Essaim de Particules », OEP) trouvent leur origine dans les *boids* de l'infographiste Craig REYNOLDS.

REYNOLDS a proposé de simuler le comportement de l'essaim d'oiseaux qui peut être compliqué à partir de trois règles simples d'application locale [3]:

1. Chaque individu doit éviter de heurter ses voisins.
2. Chaque individu tend à s'approcher des vitesses et directions générales du groupe local, c'est-à-dire des voisins immédiats.
3. Chaque individu cherche à s'approcher du centre de gravité du groupe local.

Le comportement de chaque individu à l'intérieur de l'essaim peut être modélisé par des vecteurs simples. Cette caractéristique est un des concepts de base de l'OEP.

Boyd et Richerson examinent le processus de la décision chez l'être humain, et ils ont développé le concept de connaissance et de transmission culturelle de l'individu.

D'après leurs observations, les individus ont tendance à imiter les comportements réussis qu'ils observent dans leur entourage, tout en y apportant leurs variations personnelles. Donc chaque individu formule sa décision, qui utilise ses propres expériences et les expériences des autres individus. Cette caractéristique est un autre concept de base de l'OEP [4].

2.2. DEFINTION DE L'OEP

L'OEP est une méthode d'optimisation des systèmes non linéaires développée par James KENNEDY et Russel EBERHART en 1995, c'est un modèle simplifié de relations sociales : vols d'oiseaux, bancs de poissons et essaims d'abeilles. Cette métaheuristique a été conçue surtout dans le cas continu et c'est toujours dans ce domaine que se situe la majorité des applications à ce jour. La méthode en elle-même met en jeu des groupes de particules sous forme de vecteurs se déplaçant dans l'espace de recherche. Chaque particule est caractérisée par sa position et un vecteur de changement de position appelé vitesse. En se déplaçant les particules sont influencées par leur comportement passé et par celui de leurs voisins [5]. Donc l'OEP repose sur deux règles simples [3]:

1. Chaque particule se souvient du meilleur point (le plus proche de l'objectif) par lequel elle est passée au cours de ses évolutions et tend à y retourner.
2. Chaque individu est informé du meilleur point connu au sein de la population prise dans son ensemble et tend à s'y rendre.

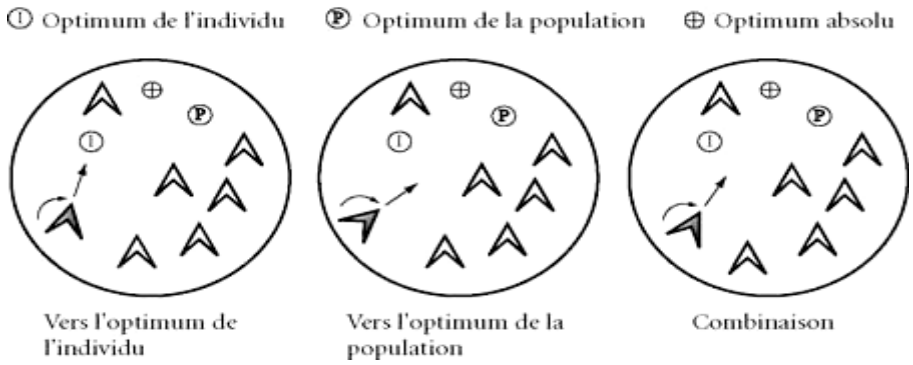


Figure 1. Principe de fonctionnement de l'OEP.

2.3. METHODE DE BASE

Etant donné un espace de recherche et une fonction sur cet espace dont on cherche l'optimum, le principe de base consiste à définir un ensemble de particules (l'essaim) par des positions et vitesses initiales, puis à le faire évoluer selon un processus raison-

nement-décision à chaque pas de temps, qui, informellement, est le suivant, du point de vue de la particule :

je me souviens de ma meilleure position jamais trouvée (meilleure position atteinte)
je demande à chacun de mes voisins leur meilleure position atteinte
je ne considère que la meilleure d'entre elles
je combine trois tendances pour définir ma nouvelle vitesse/direction de déplacement :
- ma tendance *volontariste* : à suivre mon propre chemin;
- ma tendance *conservatrice* : à revenir vers ma meilleure position atteinte;
- ma tendance *suiviste* : à suivre mon meilleur voisin.

j'effectue ce déplacement [6]

Les trois paramètres de réglage qui pondèrent les trois tendances précédentes permettent d'agir sur le compromis exploration – exploitation.

L'*exploration* est la capacité de tester différentes régions de l'espace à la recherche de bonnes solutions candidates.

L'*exploitation* est la capacité de concentrer la recherche autour des solutions prometteuses afin de s'approcher le plus possible de l'optimum [7].

2.4. FORMULATIONS

La formulation générale du mouvement d'une particule est la suivante :

$$v_i(t+1) = c_1 v_i(t) + c_2 * (p_i - x_i(t)) + c_3 * (p_g - x_i(t)) \quad (\mathbf{1})$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (\mathbf{2})$$

Où :

$v_i(t)$: la vitesse de la particule i au pas de temps t

$x_i(t)$: la position courante de la particule i au pas de temps t

p_i : la meilleure position trouvée par la particule i

p_g : la meilleure position trouvée par le voisinage de la particule i

c_1, c_2, c_3 : les coefficients de confiance pondérant les trois directions possibles (*volontariste*, *conservatrice*, *suiviste*). Dans presque toutes les applications, les coefficients c_2 et c_3 sont choisis à chaque pas de temps au hasard dans un intervalle donné [6].

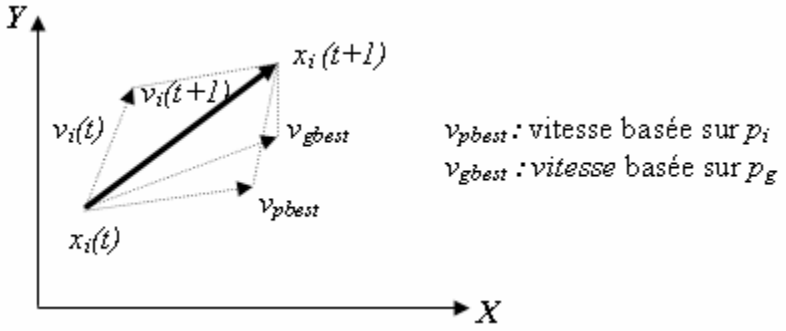


Figure 2. Concept de la modification d'une position par l'OEP.

2.5. PARAMETRES DE L'OEP

L'OEP comporte un certain nombre de paramètres que l'utilisateur est contraint de définir a priori [5]:

Taille de l'essaim (N) : qui détermine le nombre de particules impliquées à la recherche.
Dimension des particules (D) : elle est déterminée par le nombre de variables du problème.
X_{min} et X_{max} : déterminent les limites de l'espace de recherche.
V_{max} : pour limiter la vitesse d'une particule à une valeur maximale, car dans certains cas, le système pouvait exploser.
Condition d'arrêt (*MaxIterations*) : le nombre maximal d'itérations à exécuter avant de terminer la recherche.
Voisinage (*NV*) : détermine le nombre de voisinage de chaque particule.

Pour chaque particule, on définit un voisinage. Pour ce faire on fait le choix entre :

Un *voisinage physique*, qui doit être recalculé à chaque pas de temps et qui suppose l'existence d'une distance dans l'espace de recherche.
Un *voisinage social* défini une fois pour toutes.

Le voisinage le plus utilisé est celui dit « *circulaire* ». Les particules sont numérotées, disposées virtuellement sur un cercle et la définition des voisinages se fait une fois pour toutes en fonction des numéros [8].

2.6. ALGORITHME DE L'OEP

L'algorithme classique de l'OEP se résume dans les étapes suivantes [4]:

Etape 1 : Initialisation de l'essaim

La position initiale (x_i^0) ainsi que la vitesse initiale (v_i^0) de chaque particule sont généralement générées de manière aléatoire à l'intérieur du domaine de recherche

défini par [X_{min}, X_{max}]. La position courante de chaque particule sera p_i et la meilleure valeur parmi les p_i sera p_g tout en mémorisant la particule ayant cette valeur.
Etape 2 : Evaluation de la position de chaque particule
La valeur de la fonction objectif (f_i) est calculée pour chaque particule. Si cette valeur est meilleure que celle de p_i, alors cette dernière est remplacée par la nouvelle valeur. Si encore la meilleure valeur parmi les p_i est meilleure que p_g courante, alors p_g est remplacée par la nouvelle valeur calculée et la particule ayant cette valeur est mémorisée.
Etape 3 : Modification des positions
La position courante de chaque particule est modifiée en utilisant les équations (1) et (2).
Etape 4 : Vérification de la condition d'arrêt
Si le numéro de l'itération courante atteint le nombre maximum prédéterminé, alors, fin de l'algorithme. Sinon aller à l'étape 2.

Le pseudo code de l'algorithme classique de l'OEP est le suivant [5]:

```
Initialisation
La position et la vitesse initiales de chaque particule
sont initialisées aléatoirement.

Optimisation
a) Évaluation de la valeur de la fonction objectif fi
b) Si  fi meilleure que fpi Alors
     fpi =  fi , pi = xi
c) Si  fi meilleure que fpg Alors
     fpg =  fi , pg = xi
d) Mise à jour de la vitesse en utilisant l'équation (1)
e) Mise à jour de la position en utilisant l'équation (2)
f) Si la condition d'arrêt n'est pas satisfaite Alors
     revenir à (a)
   Sinon fin de l'algorithme.
```

Algorithme 1. Algorithme classique de l'OEP.

3. LE PROBLEME DU VOYAGEUR DE COMMERCE

Un voyageur de commerce doit visiter un ensemble de villes données en passant par chaque ville exactement une fois. Il commence par une ville quelconque et termine en retournant à la ville de départ où les distances entre les villes sont connues. Quel chemin faut-il choisir afin de minimiser la distance parcourue ? [9]. Une méthode d'énumération exhaustive est exclue : s'il y a N villes, pour la seconde étape de son périple notre voyageur a N-1 possibilités, pour la troisième N-2, etc. Le nombre de combinaisons en fixant la première ville, est donc (N-1)!. Pour seulement 40 villes, cela fait à peu près $2e^{46}$ solutions à tester. En supposant que l'on dispose d'un ordinateur permettant de tester un milliard de solutions par seconde, cela nous prendrait plus de $1e^{19}$ fois l'âge de l'univers pour tester toutes les solutions ! [10].

La complexité du problème croît comme la factorielle du nombre de villes. Si l'on considère des permutations simples, il existe 3 628 800 permutations de 10 villes. Pour 100 villes, on passe à 10 puissance 158 [11]. Ce problème représente la classe des problèmes NP-complets. L'existence d'un algorithme de complexité polynomiale reste inconnue. Un calcul rapide de la complexité montre qu'elle est en O(N !) où N est le nombre de villes. En supposant que le temps pour évaluer un trajet est de 1 µs, le **Tableau 1** montre l'explosion combinatoire du problème du voyageur de commerce (on considère que le nombre de possibilités est ((N-1)! /2) configurations) [12].

Nombre de villes	Nombre de possibilités	Temps de calcul
5	12	12 µs
15	43 milliards	12 heures
25	310 E+21	9,8 milliards d'années

Tableau 1. Nombre de possibilités de chemins et temps de calcul en fonction du nombre de villes (on suppose qu'il faut 1 µs pour évaluer une possibilité).

4. L'APPLICATION DE L'OEP POUR RESOUDRE LE PROBLEME

En premier, l'application exige le réglage de plusieurs paramètres utilisateur ou une configuration illustrée par les points suivants :

Les escales du voyageur de commerce, c'est-à-dire les villes qu'il doit visiter.

Le nombre de particules.

Le type de voisinage (physique ou social) :

1. Le voisinage "*physique*", qui prend en considération des distances. En pratique, les distances sont recalculées à chaque pas de temps.
2. Le voisinage "*social*", qui prend en considération "juste des rapports". En pratique, pour chaque particule, son voisinage est défini comme une liste de particules au commencement même et ne change pas. Notez que, quand le processus converge, un voisinage social devient physique.

La taille de voisinage.

La méthode à suivre pour l'exécution du processus ReHope (en cas d'aucun espoir, « NoHope »).
Aussitôt qu'il n'y a "*aucun espoir*", "*NoHope*", l'essaim est ré-étendu. Il existe deux types de processus ReHope :

1. Méthode *Simple*, l'essaim est réinitialisé simplement.
2. Méthode *Sophistiquée* [13]

   – Lazy Descent Method (LDM)
   Chaque particule va en arrière à sa meilleure position précédente et, de là, se déplace aléatoirement et lentement et s'arrête aussitôt qu'il trouve une meil-

leure position ou quand un nombre maximal de mouvements M_{max} est atteint. Si l'essaim actuel est plus petit que l'initial, il est complété par un nouvel ensemble de particules aléatoirement choisies.

- Energetic Descent Method (EDM)
  Chaque particule va en arrière à sa meilleure position précédente et, de là, se déplace lentement tant qu'il trouve une meilleure position au maximum en M_{max} mouvements. Si l'essaim actuel est plus petit que l'initial, il est complété par un nouvel ensemble de particules aléatoirement choisies.
- Local Iterative Levelling (LIL)
  Cette méthode est plus puissante et plus chère. L'idée consiste en ce que, comme il y a une infinité de fonctions objectifs possibles avec le même minimum global, nous pouvons localement et temporairement employer chacune d'entre elles pour guider une particule. Pour chaque voisin immédiat *y* (à la distance 1) de la particule *x*, une fonction objectif provisoire qui estime *f(y)* est calculée, et, selon ces évaluations provisoires, *x* se déplace au meilleur voisin immédiat.
- Adaptive ReHope Method (ARM)
  Les trois méthodes précédentes peuvent être employées automatiquement d'une façon adaptative, selon pour combien de temps (le nombre de pas de temps) la meilleure solution n'a pas été améliorée. Un exemple de stratégie est comme suit :

Nombre de pas de temps sans amélioration	**Type de processus ReHope**
0-1	No ReHope
2-3	Lazy Descent Method
4	Energetic Descent Method
>4	Local Iterative Levelling

Tableau 2. Exemple de stratégie.

Le mode d'exécution parallèle ou bien séquentiel.
L'algorithme peut s'exécuter dans le mode (simulé) parallèle ou bien dans le mode séquentiel :

- Dans le mode *parallèle*, à chaque pas de temps, des nouvelles positions sont calculées pour toutes les particules et ensuite l'essaim est généralement déplacé.
- Dans le mode *séquentiel*, chaque particule est déplacée en un pas de temps sur une voie cyclique. Ainsi, particulièrement le meilleur voisin employé au temps *t+1* ne peut pas être désormais le même comme le meilleur voisin au temps t, même si l'itération n'est pas complète.

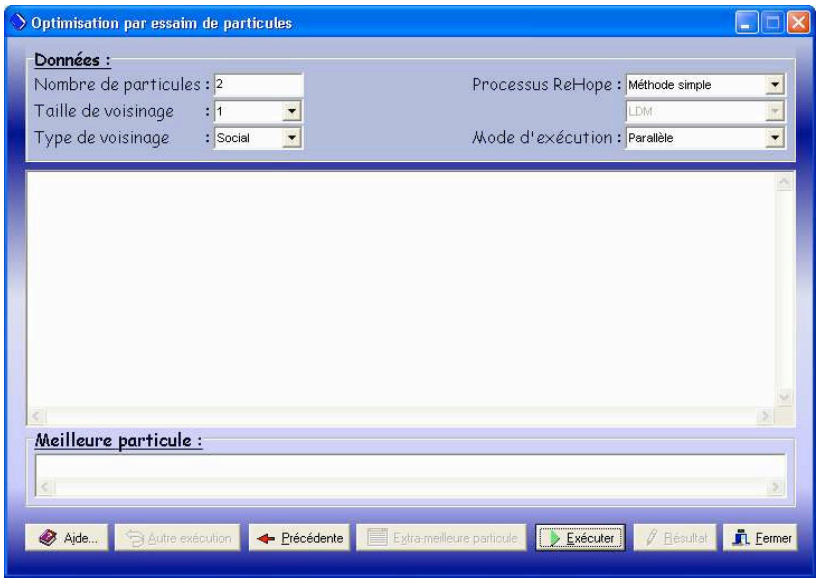


Figure 3. Une interface de configuration de l'algorithme.

Pendant l'exécution de notre application, on peut suivre l'évolution de la valeur de la fonction objectif par rapport au numéro d'itération chaque 100 itérations.

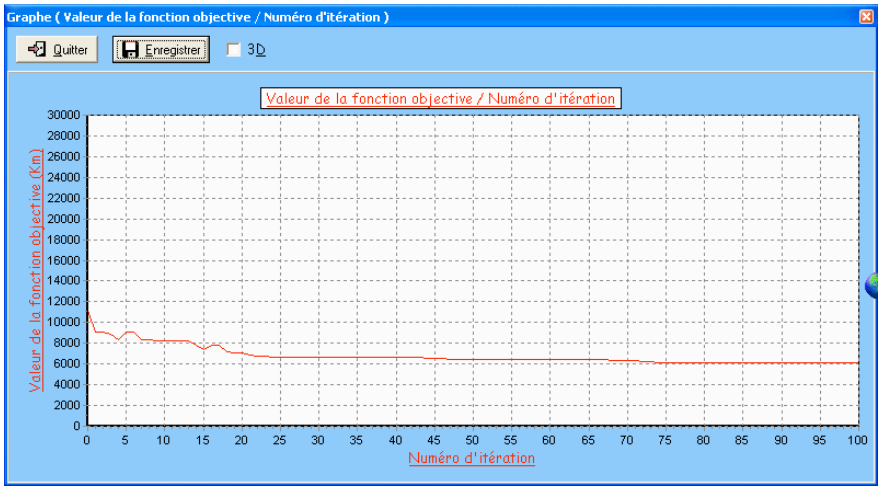


Figure 4. La valeur de la fonction objectif par rapport au numéro d'itération.

Enfin, nous avons obtenu la meilleure valeur de la fonction objectif, c'est-à-dire l'extra-meilleure particule (pour accélérer le processus, l'algorithme peut employer une extra-particule spéciale, juste pour garder la meilleure position trouvée dès le dé-

but) sous forme de trajectoire avec le parcours séquentiel des villes sur la carte géographique :

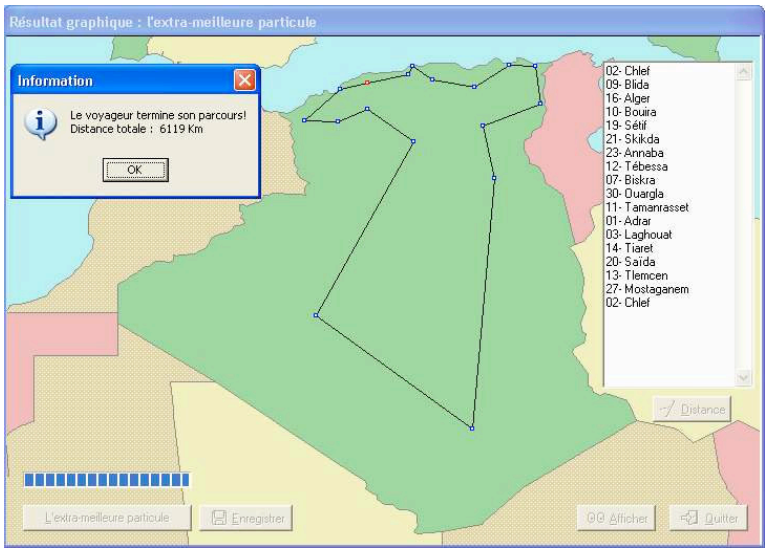


Figure 5. L'extra-meilleure particule.

Après plusieurs tests avec un graphe de 17 noeuds (villes), un nombre d'itérations limité à 10000 et des valeurs de paramètres c_2 et c_3 prises aléatoirement dans l'intervalle]0,2], les bons résultats de notre application sont obtenus par les paramètres suivants :

Nombre de particules	Taille de voisinage	Type de voisinage	C_1
30	4	Social	0.25
60	30	Social	0.25
150	4	Physique	0.25
30	15	Physique	0.5
150	15	Physique	0.25
60	30	Physique	0.25

Tableau 3. Meilleures configurations du système.

Les résultats de notre travail sont résumés comme suit :

Nombre de particules
Dans la plupart des cas, l'augmentation du nombre de particules dans l'essaim permet d'atteindre la meilleure solution car le taux de succès de l'algorithme a été également amélioré par l'augmentation du nombre de particules. Cela est normal, puisque plus de particules échantillonnent mieux l'espace de recherche.

Taille de voisinage
Habituellement, si on prend la taille de voisinage égale à la moitié du nombre de particules, l'algorithme donne un bon résultat.

Type de voisinage
Après le calcul de la distance moyenne de nos tests, on a trouvé que dans le cas du voisinage physique la distance moyenne est D_{ph} = 6864.28 Km mais par contre, dans le cas du voisinage social la distance est D_{so} = 6940.83 Km et cela se signifie que $D_{ph} < D_{so}$ mais cette différence est négligeable car l'utilisation du voisinage physique est beaucoup plus chère (en terme du temps de calcul) que le voisinage social.

Paramètre c_1
A la lecture du **Tableau 3**, indique que la meilleure valeur du coefficient c_1 est égale à 0.25 car dans la plupart des cas, l'utilisation de cette valeur permet d'obtenir de bons résultats.

5. CONCLUSION ET PERSPECTIVES

Dans ce papier, nous avons fait une étude sur l'Optimisation par Essaim de Particules (OEP) comme une nouvelle méthode inspirée du comportement social des individus qui ont tendance à imiter les comportements réussis qu'ils observent dans leur entourage, tout en y apportant leurs variations personnelles. D'après cette étude, nous avons constaté que:

- L'optimisation par essaim de particules est un modèle très intéressant pour la résolution de différents problèmes d'optimisation grâce à ses diverses variations, chacune dédiée à un type spécifique de problèmes.
- Elle peut être appliquée avec succès pour résoudre des problèmes d'optimisation du monde réel de l'ingénierie et de télécommunication, entre autre des problèmes d'optimisation multidimensionnelle, l'entraînement des réseaux de neurones artificiels et des problèmes d'optimisation multiobjectif.
- L'OEP utilise une population de particules, chacune a une petite mémoire, lui permettant de se souvenir de sa meilleure performance, en position et en valeur. La relation entre ces particules est fondée sur le principe de la collaboration.
- L'algorithme de l'OEP est très simple à appliquer, après avoir défini les éléments de base suivants : l'espace de recherche, position d'une particule, vitesse d'une particule, voisinage d'une particule et aussi les opérateurs (l'addition d'une position avec une vitesse, la somme de deux vitesses, la différence entre deux positions, le produit d'un coefficient de confiance par une vitesse) des équations de base de l'OEP.
- L'algorithme de l'OEP comprend plusieurs paramètres de réglage qui permettent d'agir sur le compromis exploration / exploitation.

Ceci nous a encouragés à explorer davantage ce modèle et d'essayer de l'exploiter pour résoudre le problème du voyageur de commerce, le modèle d'optimisation combinatoire le plus utilisé dans la littérature.

Notre algorithme a été largement testé et les résultats expérimentaux montrent :

- Que l'OEP est efficace pour la résolution de problèmes d'optimisation combinatoire.
- Ainsi, nous sommes arrivés à identifier les valeurs optimales des paramètres qui assurent les bons résultats.

REFERENCES

[1]: Joël QUINQUETON, Aspects socio-organisationnels dans les systèmes multi-agents : l'intelligence artificielle en essaim, FING 2004.

[2]: Wikipédia : l'encyclopédie libre, Optimisation par essaims particulaires, http://fr.wikipedia.org/wiki/optimisation_par_essaims_particulaires, 29 Novembre 2006.

[3]: Jean-Philippe RENNARD, Vie artificielle, http://www.futura-sciences.com, 2/10/2003.

[4]: Y. FUKUYAMA, Fundamentals of particle swarm optimization techniques, IEEE PES Tutorial on Modern Heuristic Optimization Techniques with Application to Power Systems, Chapter 5, jan.28-feb.1 2002.

[5]: Hassiba TALBI & Mohamed BATOUCHE, L'optimisation par essaim de particules : Application au recalage d'images, Séminaire SNIB, 05 Mai 2004.

[6]: Maurice CLERC, The Swarm and the Queen : Towards a Deterministic and Adaptive Particle Swarm Optimization, Congress on Evolutionary Computation, Washington D.C., IEEE, 1999.

[7]: Ioan Cristian TRELEA, L'essaim de particules vu comme un système dynamique : convergence et choix des paramètres, Séminaire OEP, Paris, Carré des Sciences, 02 Octobre 2003.

[8]: Maurice CLERC, L'optimisation par essaim particulaire, Tutoriel pour OEP 2003 (partie I).

[9]: AIT-AHMED Madjid & LEBORGNE Fabien, Heuristique pour le problème du voyageur de commerce, Manuel utilisateur, Décembre 2003.

[10]: V. MAGNIN, AG et voyageur de commerce (applet Java), http://www.eudil.fr/~vmagnin/coursag/voyageur/voyageur.html, 29 Janvier 2003.

[11]: Jean-Philippe RENNARD, Vie artificielle, http://www.rennard.org/iva/ agapll.html, 17/01/2003.

[12]: TOLLARI Sabrina, Le problème du voyageur de commerce, http://sis.univ-tln.fr/~tollari/TER/AlgoGen1/node4.html, 23-05-2003.

[13]: Maurice CLERC, Discrete Particle Swarm Optimization Illustrated by the Travelling Salesman Problem, 29 Février 2000.

Modèle déformable appliqué à la détection du contour d'images médicales

M. DEBAKLA [(1)],M.BENYETTOU [(2)], A.BOUADI [(2)]

[(1)] centre universitaire Mustapha stambouli Mascara
[(2)] Université Mohamed Boudiaf USTO Oran

debakla_med@yahoo.fr, **med_benyettou@yahoo.fr**, **bouadi_abed@yahoo.fr**

Abstract. L'imagerie médicale est un outil de diagnostic et de suivi primordial, et l'interprétation des données qu'elle peut fournir est un élément important du dossier patient. L'exploitation de ces images nécessite un traitement de bas niveau tel que la segmentation. La qualité de la segmentation, c'est à dire, la précision de la localisation et la non-confusion des régions ont un impact direct sur les performances de l'application. Les méthodes de segmentation d'images sont nombreuses; toutes présentent des avantages mais ne donnent pas entière satisfaction. Toute doivent être adaptées en fonction des applications que l'on se propose de réaliser. Les contours actifs ou modèles déformables ont permis de s'affranchir du chaînage des points du contour mais nécessitent le réglage de nombreux paramètres. Dans ce travail nous avons étudié l'influence des paramètres des différentes énergies du snake sur les résultats en utilisant l'algorithme « greedy ». Pour voir l'importance de ces paramètres, de nombreux tests ont été réalisés sur différents types d'images (artificielles et réelles), avec différents jeux de coefficients des énergies.

Mots-clefs : Imagerie médicale, segmentation, traitement d'images, contours actifs, algorithme greedy.

1 Introduction

Les contours actifs tirent leur origine des modèles élastiques [Bur 81], mais la communauté s'accorde à les attribuer à l'équipe Kass, Witkin et Terzopoulos [Kas 87], [Kas 88] qui introduisirent les snakes ou courbes minimisantes. Les snakes tiennent leur nom de leur aptitude à se déformer comme des serpents. Depuis la publication de cette équipe, les modèles déformables sont devenus un sujet très important pour la communauté du traitement d'images. De très nombreuses équipes s'y sont intéressées de manière plus ou moins approfondie. [Bal 99], [Bal 02],
Les domaines d'utilisation sont nombreux tant en 2D qu'en 3D tels : la reconnaissance de formes, le suivi de scènes, la segmentation d'images. [Coh 92], [Coh 96],

M. DEBAKLA (1),M.BENYETTOU (2), A.BOUADI (2)

Un contour actif est encore de nos jours difficile à utiliser pour la détection de contour. Celui-ci requière l'ajustement de plusieurs paramètres suivant les images à traitées et ne peut donc être facilement exploité par des personnes inexpérimentées.
Le domaine qui nous intéresse plus particulièrement est celui de l'imagerie médicale. Les sources d'images sont diverses. Le traitement d'images peut servir à automatiser des tâches usuellement accomplies par les médecin à l'aide de ses traditionnelles règles spécialisées.
Dans ce travail nous avons étudié l'influence des paramètres des différentes énergies du snake sur les résultats en utilisant l'algorithme « greedy ». Pour voir l'importance de ces paramètres de nombreux tests ont été réalisés sur différents types d'images, avec différents jeux de coefficients des énergies.
L'enjeu sera donc de déterminer si la modification des coefficients d'énergie suivant leur influence dans le choix du déplacement d'un contour actif, améliore la détection des contours d'une image. L'étude sera réalisée sur le contour actif à coefficients dits constants.
Dans la première partie de ce travail nous présentons les bases de la méthode des contours actifs et l'algorithme greedy. La deuxième partie est consacrée à l'étude des tests de notre implémentation sur des images médicales ou nous utilisons un jeu d'essai sur les coefficients d'énergie.

2 Définitions [Rou 03] [Coh 92]

Les contours actifs sont modélisés par une courbe continue, fermée ou non, à extrémités fixes ou non et qui se déforme en fonction de différentes énergies pour venir se positionner sur les frontières de l'objet. Les contours actifs sont devenus des surfaces actives, tout en gardant le principe de la déformation qui doit minimiser une énergie ***E***

$$E = E_{int} + E_{ext} + E_{cont} \quad (1)$$

Avec
E_{int} les énergies internes.
E_{ext} les énergies externes
E_{cont} les énergies de contexte

Les énergies internes gèrent la cohérence de la courbe. Elles définissent la raideur de la courbe et la cohésion des points.
Les énergies externes prennent en compte les caractéristiques des images traitées.
Enfin, les énergies de contexte, permettent d'introduire des connaissances a priori sur ce que nous cherchons.

Modèle déformable appliqué à la détection du contour d'images médicales

Etant donné l'existence de différentes formes d'images, des coefficients sont attribués à chacune des énergies utilisées. La fonctionnelle d'énergie représentera alors l'ensemble des énergies, auxquelles sont appliqués leur coefficient respectif. L'approche tendra à minimiser cette fonctionnelle d'énergie.
Le contour est défini par une représentation paramétrique.

$$v : \Omega = [0,1] \rightarrow \square^{2}$$

Soit *C* une courbe déformable spatialement et temporellement [Del 00]. Le domaine de définition de la courbe est normalisé à $[0,1]$ et le paramètre est l'abscisse curviligne. Avec *s* le paramètre de position spatiale le long de la courbe dans l'image et *t* l'évolution temporelle de la courbe dans, la courbe se représente comme sur la figure1.

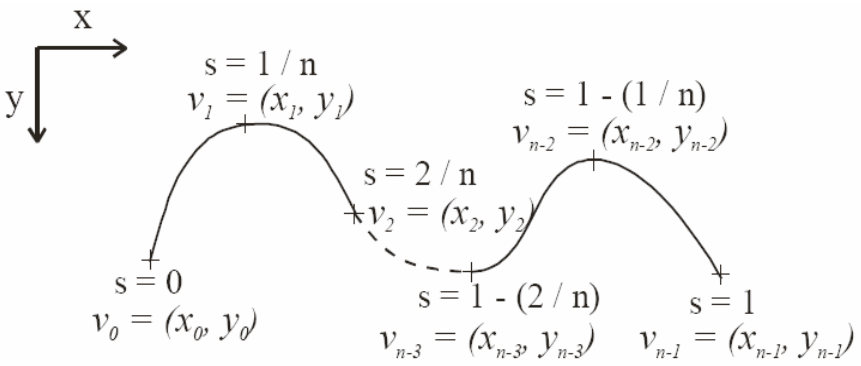


Fig. 1. *Exemples de coordonnées cartésiennes et abscisses curvilignes d'un contour actif*

Soient *v(s, t)* la position d'un point de la courbe à une instante *t* et *x, y* les coordonnées cartésiennes d'un point de l'image.

$$v(s,t) = {}^{t}(x(s,t), y(s,t)) \forall (s,t) \in [0,1] \times [0,+\infty] \quad (2)$$

Cette courbe est déformée de manière itérative afin de minimiser une fonctionnelle d'énergie.

M. DEBAKLA (1),M.BENYETTOU (2), A.BOUADI (2)

3 Description des énergies

3.1 Energie interne

L'énergie interne est calculée à partir de deux forces appelées respectivement continuité et courbure :

$$\boldsymbol{E_{int} = E_{con} + E_{cou}}$$

E_{con} *et* E_{cou} sont les énergies associées respectivement aux forces de continuité et courbure.

3.1.1 L'énergie de Courbure

Le but de cette énergie est d'éviter que le contour contienne des points isolés qui ne seraient pas cohérents avec la forme globale régulée du contour. L'énergie de courbure définit l'angle formé par trois points adjacents du snake. Plus le coefficient attribué à cette énergie sera fort, et plus le snake aura la forme connue pour être la forme la moins « coûteuse » en énergie : la ligne droite ou le cercle pour des objets fermés.

L'énergie de courbure est donnée par la relation suivante :

$$E_{cou}(n_i) = 0.5((Xpo\,\mathrm{int}(v_{i-1}) - 2Xpo\,\mathrm{int}(n_i) + Xpo\,\mathrm{int}(v_{i+1}))^2 \quad (3)$$
$$+(Ypo\,\mathrm{int}(v_{i-1}) - 2Ypo\,\mathrm{int}(n_i) + Ypo\,\mathrm{int}(v_{i+1}))^2)$$

3.1.2 L'énergie de Continuité

L'énergie de continuité influe sur le rayon de courbure du contour en conduisant les points du contour à se positionner de manière à être équidistants. La forme du contour tend alors vers un cercle.

L'énergie de continuité se calcule par rapport à la moyenne des distances entre les points du snake, telle que :

$$Econ(n_i) = 0.5\,abs(Distance Moyenne - \quad (4)$$
$$\sqrt{(Xpo\,\mathrm{int}(v_{i-1}) - Xpo\,\mathrm{int}(n_i))^2 + (Ypo\,\mathrm{int}(v_{i-1}) - Ypo\,\mathrm{int}(n_i))^2}$$

Modèle déformable appliqué à la détection du contour d'images médicales

3.2 Energie externe

L'énergie externe correspond à l'adéquation aux données. Cette énergie externe prend en compte les caractéristiques de l'image. Rappelons ici que ce sont les contours de formes qui sont recherchés donc les points de fort gradient ou des points ayant une propriété de position par rapport à une couleur donnée.
L'énergie externe utilise deux forces qui sont liées aux gradients couleur et à l'intensité de gris de pixel :

$$E_{ext} = E_{int} + E_{gra}$$

Où E_{gra} et E_{int} sont les énergies représentant respectivement les gradients couleur et l'intensité de pixel de image.

3.2.1 L'énergie d'intensité

L'énergie d'intensité représente simplement la valeur du pixel (sur un octet, donc de 0 à 255) de l'image transformée en niveaux de gris.

$$Eint = Niveaux\ de\ Gris[x][y] \quad (5)$$

3.2.2 L'énergie de Gradient

L'énergie de gradient est d'une importance première pour la détection du contour. En effet, un contour est généralement caractérisé par une forte différence entre les valeurs de plusieurs pixels.
Ainsi, si la dérivée d'une fonction représente les pentes d'une courbe, le gradient montrera les fortes différences, les contours de l'image.
En mathématiques, le gradient d'une fonction spatiale (dans un graphe à trois dimensions au minimum) représente simplement les dérivées partielles de celle-ci.

$$\overrightarrow{grad}(f(x,y)) = \frac{\partial f}{\partial x}\vec{i} + \frac{\partial f}{\partial y}\vec{j} \quad (6)$$

En informatique, les fonctions sont discrètes, et il devient difficile voir impossible d'appliquer cette formule. Aussi, pour obtenir le gradient d'une image, donc d'une matrice de pixels, nous pouvons utiliser les méthodes de Roberts, Prewitt ou de Sobel.

3.3 Energie de contexte [Rou 03]

L'énergie de contexte, parfois appelée énergie de contrainte, permet d'introduire des connaissances a priori sur ce que nous cherchons. Entre autres, nous plaçons, sous cette rubrique, l'énergie ballon introduite par Laurent D. Cohen [Coh 91].
L'énergie de Ballon est l'énergie qui décide du sens de propagation du contour actif. Un coefficient d'énergie de ballon positif va concentrer le snake, alors qu'un coefficient négatif va rendre le snake expansif.
Pour calculer l'énergie de Ballon, nous étudions la distance entre un point du snake et le barycentre des différents points composant le snake.

$$E_{bal(ni)} = ((Xpo\,\mathrm{int}(n_i) - moyenneX).(Xpo\,\mathrm{int}(n_i) - Xpo\,\mathrm{int}(v_i))) \quad (7)$$
$$+((Ypo\,\mathrm{int}(n_i) - moyenneY).(Ypo\,\mathrm{int}(n_i) - Ypo\,\mathrm{int}(v_i)))$$

C'est à partir de ces différentes énergies et de leur combinaison que l'on peut définir un problème dont la solution est le contour recherché. La mise en oeuvre d'une telle approche a donné lieu à de nombreuses implémentations .Parmi ces implémentations l'algorithme « greedy ».

4 L'Algorithme greedy [Rou 03]

L'algorithme g*reedy ou glouton* décrit par Williams et Shah [Wil 92] est utilisé pour minimiser l'énergie d'un contour actif, Williams et Shah, de manière classique, discrétisent l'expression :

$$E(c) = \sum_{t=1}^{N} aE\,\mathrm{int}\,ensité(M_i) + cEcontinuité(M_i) + bEcourbure(M_i) \quad (8)$$
$$+ gEgradient(M_i) + dEballon(M_i)$$

Avec M_i, *i=1..N,*un ensemble de points ordonnés et a,c,b,g,d les coefficients attribués à chaque énergie

La méthode du greedy consiste à faire évoluer le contour actif en minimisant la fonctionnelle d'énergie : pour chaque point du snake, on choisit un nombre de voisins pour lesquels on va calculer l'énergie, le point se déplacera alors sur le voisin qui

Modèle déformable appliqué à la détection du contour d'images médicales

possède l'énergie la plus faible. On cherche donc l'ensemble *M* des points pour laquelle l'énergie est minimale.

C'est donc un algorithme itératif qui déplace un point unique pour constituer un nouveau contour actif à chaque itération. Tous les points sont traités successivement lors de chaque itération.

Dans l'algorithme g*reedy* utilisé, la fonctionnelle d'énergie est une somme de cinq énergies (intensité, continuité, courbure, gradient et ballon) normalisées et pondérées chacune par un coefficient (a, b, c, g et d).

5 Implémentation et résultats

Nous avons implémenté un snake classique basé sur l'algorithme ***greedy.***

5.1 Initialisation du contour actif

Pour une image donnée, l'initialisation par un cercle est effectuée par la détermination du rayon et le point du centre de coordonné (a, b). Tout d'abord un cercle est représenté par une équation géométrique $(x - a)2 + (y - b)2 = r2$. Sa position et sa taille sont déterminées de telle sorte soit qu'il englobe l'objet qu'on veut segmenter par le contour actif ou bien qu'il se positionne à l'intérieur de cet objet.

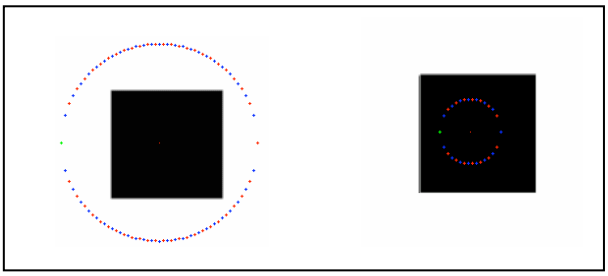


Fig. 2. Initialisation du snake par un cercle

5.2 Les énergies utilisées

La fonctionnelle d'énergie attachée au contour actif utilisé est donnée par :

$$E_{totale} = cE_{continuité} + bE_{courbure} + gE_{gradient} + aE_{intensité} + dE_{ballon}$$

5.3 Sensibilité des paramètres

Dans notre application, nous nous plaçons par rapport à une démarche classique qui consiste à régler par essai-erreur les pondérations de la combinaison linéaire d'énergies mise en oeuvre dans un contour actif. Nous avons cherché à déterminer

quelle est la sensibilité des résultats à ces paramètres (figure 3). L'instabilité liée aux valeurs de ces coefficients montre la difficulté qu'il y a à déterminer les valeurs optimales. Sur différentes images (artificielles, réelles), nous faisons évoluer un contour actif. Le modèle met en oeuvre les énergies de continuité, de courbure, du gradient, de ballon, et d'intensité.
Un jeu de paramètres a été retenu après de multiples essais. Nous avons alors fait varier chacun des paramètres d'une valeur très faible en plus ou en moins de la valeur idéale. Nous montrons, sur quelques exemples, les points que l'on peut considérer comme des points inexacts sur le contour final

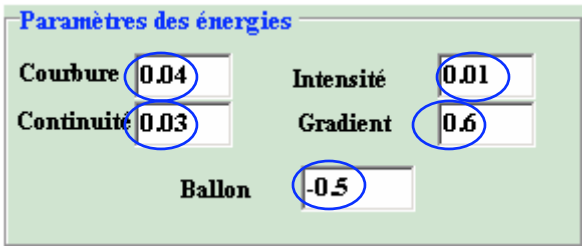


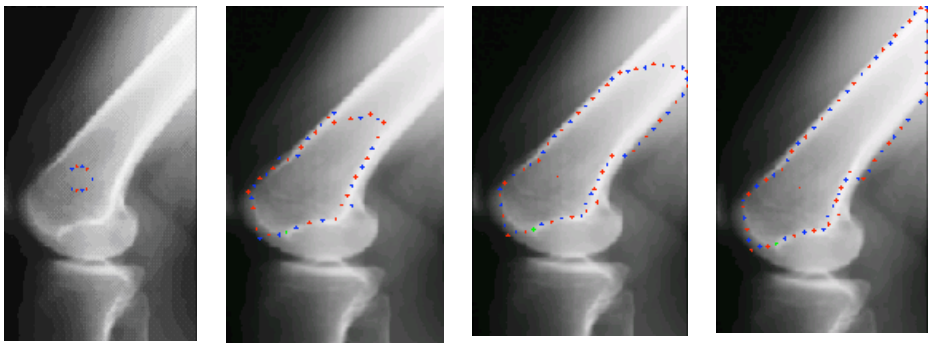
Fig. 3. Les coefficients (a,b,c,d,g) attribués aux énergies

.

. .Il en ressort que tous les paramètres ont une influence très sensible. Sur une radio d'un condyle, nous allons voir la grande sensibilité des résultats à ces paramètres.

Paramètres retenus

b	c	a	g	d	X	Y	rayon	Nb points	Nb itération
0.7	0.45	0.1	0.98	-0.65	60	140	10	10	261

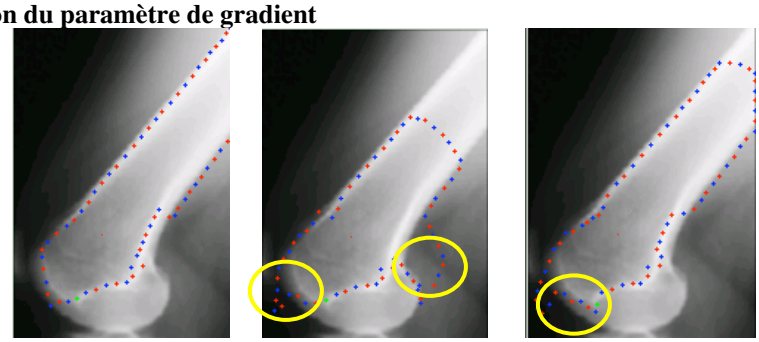


a)Initialisation b) 74 itérations c) 168 itérations d) 261 itérations

Fig. 4. Déformation du snake jusqu'à trouver le contour cherché

Modèle déformable appliqué à la détection du contour d'images médicales

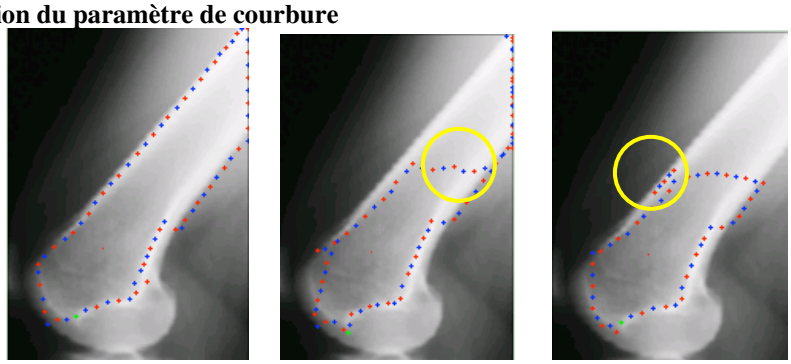
Variation du paramètre de gradient



g retenu: 0.98 **g:** 0.9 **gt :** 0.96

Fig. 5. *Influence de la variation d'un paramètre (gradient)*

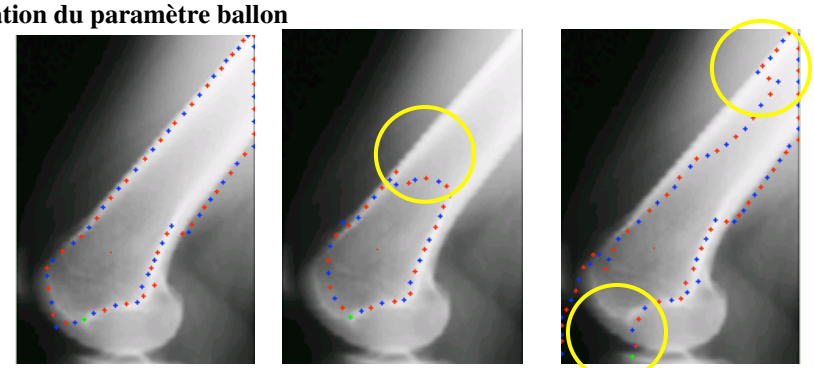
Variation du paramètre de courbure



b retenu : 0.7 **b :** 0.6 **b :** 0.86

Fig. 6. Influence de la variation d'un paramètre (courbure)

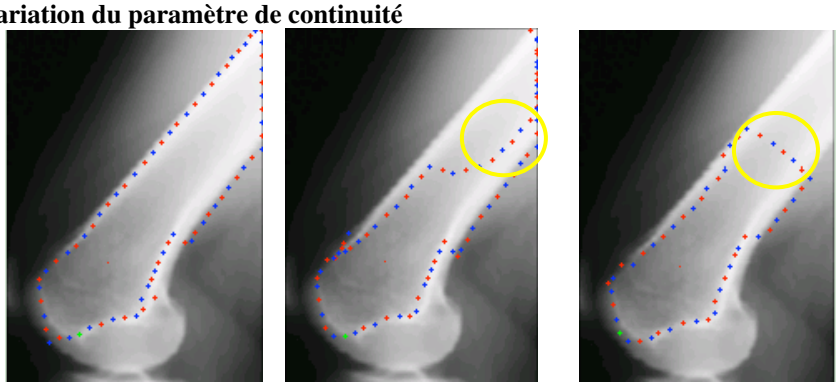
Variation du paramètre ballon



d retenu : -0.65 **d :** -0.5 **d :** -0.7

Fig. 7. *Influence de la variation d'un paramètre (ballon)*

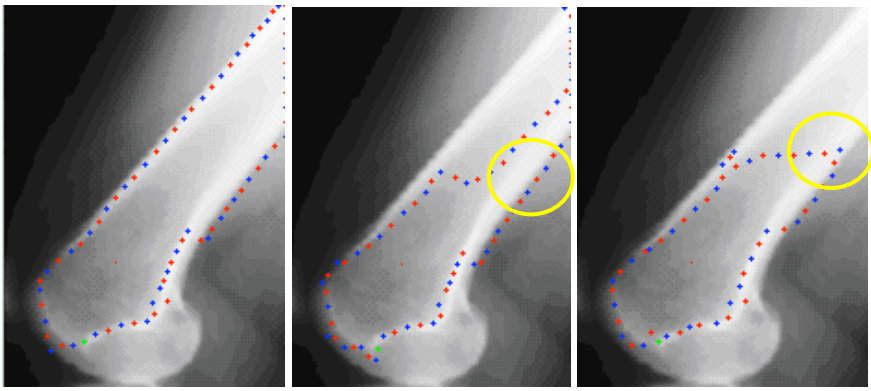
Variation du paramètre de continuité



c retenu : 0.45 **c :** 0.4 **c :** 0.5

Fig. 8. Influence de la variation d'un paramètre (continuité)

Variation du paramètre d'intensité



a retenu : 0.1 **a :** 0 **a :** 0.11

Fig. 9. Influence de la variation d'un paramètre (intensité)

5.4 Résultats obtenus sur des images médicales

Nous avons entrepris différents tests sur trois images : le fémur, l'épaule et le crâne. Cependant, la réalisation de ces tests demandait de connaître au départ des coefficients d'énergie assez bons, et cela a été difficile d'en obtenir avec un seul test. Voici la synthèse des résultats sur trois images, mentionnant toutes les informations nécessaires à la reproduction des essais

Modèle déformable appliqué à la détection du contour d'images médicales

1 Le femur

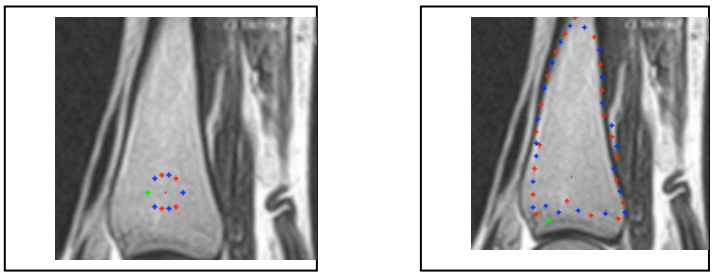


Fig. 10. *Le fémur, lors de l'initialisation des points et à la fin des itérations.*

Avec les paramètres suivants

b	c	a	g	d	X	Y	rayon	Nb points	voisinage
0.65	0.6	0.09	1	-0.4	120	100	10	10	1

Critère d'arrêt

92 %des points du snake sont immobiles pendant 5 itérations

2 L'épaule

Fig. 11. L'épaule, lors de l'initialisation des points et à la fin des itérations.

Avec les paramètres suivants

b	C	a	g	d	X	Y	rayon	Nb points	voisinage
0.5	0.5	1	1	-0.5	55	58	6	6	1

Critère d'arrêt

95% des points du snake qui ne bougent pas pendant 3 itérations

3 Le crâne

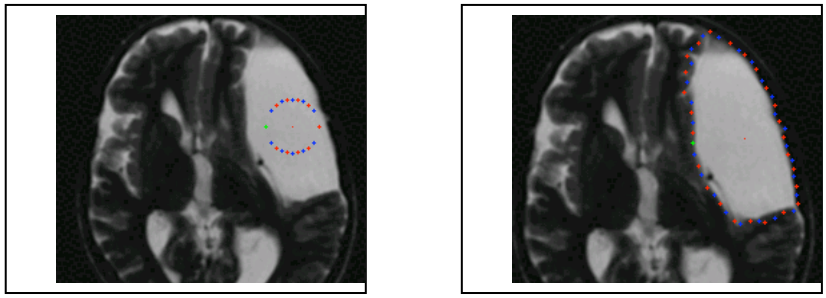


Fig. 12. Le crâne, lors de l'initialisation des points et à la fin des itérations.

Avec les paramètres suivants

b	c	a	g	d	X	Y	rayon	Nb points	voisinage
0.5	0.55	0.02	1	-0.1	190	90	20	20	3

Critère d'arrêt

90% des points du snake qui ne bougent pas pendant 3 itérations

6 Conclusion

Dans ce travail, nous nous sommes intéressés à l'application du contour actif à la détection des contours d'objets dans des images médicales. Le choix des snakes nous a permis d'éviter d'utiliser un algorithme suiveur de contours. Nous avons obtenu de bons résultats sur des images médicales. D'après cette implémentation nous avons remarqué que les contours actifs sont sensibilité au bruit, difficulté de réglage des différents paramètres, problème d'initialisation du snakes et ils sont influencés par des propriétés des images. En effet, la mise en oeuvre de contours actifs par un utilisateur non expert s'avère délicate car le modèle doit être choisi et réglé en fonction du type d'images et du type d'objets présents dans l'image. Vu cette complexité de nombreuse amélioration sur le snake classique ont été apporté citons à titre d'exemples le modèle géodésique [Cas 97], , Les B-snakes [Bri 00], le modèle particulaire[Ang 01], [Par 01] [Rou 04]

Références

[Ang 01] ANGELLA, F. - *Modèles déformables et Systèmes particulaires : application à l'extraction de structures arborescentes en analyse d'images*. Thèse de doctorat de l'université de Bordeaux 1, spécialité : Automatique, Productique, Signal et Image, 9 janvier 2001, p. 131.

Modèle déformable appliqué à la détection du contour d'images médicales

[Bal 99] BALLERINI, L. " Genetic Snakes for Medical Images Segmentation. *Lectures"Notes in Computer Science*, vol. 1596, ISSN 0301-9743, 1999, p. 59-73.

[Bal 02] BALLERINI, L. BOCCHIO, L. "*Segmentation of Liver Images by Texture and Genetic Snake"s*. 2nd European Medical and Biological Conference, Vienne, Autriche, 4-8 décembre 2002.

[Bri 00] BRIGGER, P. HOEG, J. UNSER, M. "B-Spline Snakes: A Flexible Tool for Parametric Contour Detection". *IEEE Transaction on Image Processing*, vol. 9, n° 9, septembre 2000, p. 1484-1496.

[Bur 81] BURR, D.J. "Elastic Matching of Line Drawings". IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI), vol. 3, n° 6, novembre 1981, p. 708-713.

[Cas 97] CASELLES, V KIMMEL, R. SAPIRO, G. "Geodesic Active Contours". *International Journal of Computer Vision*, vol. 1, n° 22, 1997, p. 61-79.

[Coh 92] COHEN, I. « *Modèles Déformables 2-D et 3-D : Application à la Segmentation d'Images Médicales »*. Thèse de doctorat de l'Université de Paris IX - Dauphine, spécialité : Mathématiques appliquées, 03 juin 1992, p.152. Disponible le 01/11/2001 sur http://www.inria.fr/rrrt/index.fr.html

[Coh 96] COHEN I, HERLIN I. "Optical Flow and Phase Portrait Methods for Environnemental Satellite Image Sequences", ECCV 1996, Oxford, UK, 1996.Disponible 23/01/2002 sur: http://www-air.inria.fr/Publications/abstracts/eccv96.html.

[Kas 87] KASS, M. WITKIN, A. TERZOPOULOS, D. " *Snakes: Active Contour Models"*, In Proceedings of the first International Conference on Computer Vision, juin 1987, p.259-268.

[Kas 88] KASS, M. WITKIN, A. AND TERZOPOULOS D "Snakes: Active Contour Models. *International Journal of Computer Vision"*, vol. 55, 1988, p. 321-331

[Par 01] PARK, J. KELLER, J.M. - Snakes on the Watershed. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, n° 10, octobre 2001, p. 1201-1205.

[Rou 03] ROUSSELLE Jean-Jacques. " Les contours actifs, une méthode de segmentation application à l'imagerie médicale", thèse de Doctorat de l'université François Rabelais de Tours Le 9 juillet 2003

[Rou 04] ROUSSELLE Jean-Jacques NEBEL Fabien "Amélioration des paramètres des contours actifs par une technique issue de la théorie de l'apprentissage". Rapport de stage, 2002-2004 de l'université François Rabelais de Tours.

Isometric embeddings of subdivided connected graphs

Méziane AÏDER[1], Sylvain GRAVIER[2] and Kahina MESLEM[1]

[1] Faculté de Mathématiques, U.S.T.H.B, BP.32 El-Alia, 16111 Alger, Algerie
[2] CNRS, Institut Joseph Fourier. 100, Rue des Maths BP 74. 38 402. Saint Martin d'Hères Grenoble Cedex, France
m-aider@usthb.dz, gravier@fourier.ujf-grenoble.fr, kahina_meslem@hotmail.com

Abstract. Isometric subgraphs of hypercubes are known as partial cubes. These graphs have first been investigated by Graham and Pollack [4], and Djokovič [3]. Several papers followed with various characterizations of partial cubes. In this paper, we determine among all subdivisions of a given connected graph which ones are partial cubes. We also prove that that a subdivision of a complete graph of order K_5, is a partial cube if and only if this one is isomorphic to $S(K_5)$, or there exist non-subdivided edges of K_5 adjacent to a common vertex in the subdivision and the other edges of K_5 are subdivided an odd number of tim

1 Introduction

For a graph G, the distance $d_G(u, v)$ between vertices u and v is defined as the number of edges on a shortest uv-path. A subgraph H of G is called isometric if and only if $d_G(u, v) = d_H(u, v)$ for all $u, v \in V(H)$. Isometric subgraphs of hypercubes are called *partial cubes*. Partial cubes have first been investigated in 1970's by Graham and Pollak [4] where they used a model for a communication network. Later, partial cubes have been drawn attention of several other authors who provide the characterization and the algorithmic aspect of partial cubes. In [5], the recognition of such class have been solved in polynomial time. However Djokovič [3] has given an important characterization of partial cubes using a relation defined in the edges set.

Clearly, partial cubes are bipartite. The simple way to obtain a bipartite graph is to subdivide every edge of G by a single vertex. Such a graph is a subdivision of G and denoted $S(G)$. A subdivision of a graph G is a graph obtained from G by adding vertices to its edges. In the characterization context, the subdivision of the wheel has been treated as partial cubes and important results were provided. The subdivided wheels result was interesting since it consists in answering in negative a question of Chepoi and Tardif [2] whether partial cubes are precisely bipartite graphs with convex intervals. However, the main question that we have treated, is how to determine which subdivision is a partial cube. Consequently, in this paper, we determine among all the subdivisions of a given configuration, which ones are partial cubes. In this context, we characterize the

subdivisions of a diamond $K_4 - e$ (a fan F_k respectively) which are partial cubes. Being interested in studying the subdivision of K_5, we come to notice important properties of complete graph K_n subdivisions.

2 Preliminaries definitions

Let $G = (V, E)$ be a graph where V is its vertices set and E is its edges set. A *walk* is a sequence of vertices $v_1, v_2, ..., v_n$ and edges $v_i v_{i+1}$, $1 \leq i \leq n-1$. G is a complete graph if it is isomorphic to a graph where any two distinct vertices are adjacent. For any edge e in $E(G)$, $G - e$ is the resulting graph by deleting the edge e. A *path* on n vertices is a walk on n different vertices and denoted P_n. A closed walk in which all vertices (except the first and the last) are different is a *cycle*. The cycle on n vertices is denoted C_n. A *chord* of a cycle C_n is any edge connecting two non consecutive vertices. For a graph G, the *distance* $d_G(u, v)$ (or briefly $d(u, v)$), between vertices u and v is defined as the number of edges on a shortest uv-path. A subgraph H of G is called *isometric* if $d_G(u, v) = d_H(u, v)$ for all $u, v \in V(H)$.

The vertex set of the *n-cube* (or the *hypercube*) Q_n consists of all n-tuples $b_1, b_2, ..., b_n$ with $b_i \in \{0, 1\}$, where two vertices are adjacent if the corresponding tuples differ precisely in one place. Q_n is a bipartite graph. An isometric subgraph of Q_n is called *partial cube*. A graph G is an *isometric embedding* of the hypercube if it is isomorphic to a partial cube.

A *subdivision* of a graph G, noted $sub(G)$, is the graph obtained from G by adding vertices to the edges of G. $S(G)$ is a subdivision of a graph G where each edge of G contains exactly one added vertex. In [9], Klavzar and Lipovec showed that $S(G)$ is partial cube if and only if every block of G is either a cycle or a complete graph. A vertex v in G which is adjacent to all its neighbors of G in $sub(G)$ is said *universal* in $sub(G)$. That means that all the edges of G adjacent to v are not subdivided.

Two edges $e = xy$ and $f = uv$ of a connected graph are in the Djokovic-Winkler [3] [10] relation θ if $d_G(x, u) + d_G(y, v) \neq d_G(x, v) + d_G(y, u)$. In symbols, $e\theta f$. It's easy to see that in a shortest path P of a graph G, no two distinct edges are in relation θ. In an isometric cycle C_{2k}, each edge e is in relation exactly one different edge f with respect to the relation θ. If G is a bipartite graph, the edges $e = xy$ and $f = uv$ are in relation θ then $d(u, x) = d(v, y) = d(u, y) - 1 = d(v, x) - 1$. Winkler [10] proved that a bipartite graph is a partial cube if and only if the relation θ is transitive.

Let G' be a connected graph. A *proper cover* G'_1, G'_2 consists of two isometric subgraphs G'_1, G'_2 such that $G' = G'_1 \cup G'_2$ and $G'_0 = G'_1 \cap G'_2$ is a nonempty subgraph, called *intersection* of the cover. The *expansion* of G' with respect to G'_1 and G'_2 is the graph G constructed as follow. Let G_i be an isometric copy of for $i = 1, 2$ and any vertex u', let u_i be the corresponding vertex in G_i, for $i = 1, 2$. Then G is obtained from the disjoint union $G_1 \cup G_2$, where for each $u' \in G'_0$, the vertices u_1 and u_2 are joined by an edge. Chepoi [2] proved that a

graph G is a partial cube if and only if it can be obtained from K_1 by a sequence of expansions.

3 Partial cubes as subdivisions of connected graphs

If T is a tree, then $sub(T)$ is also a tree and thus a partial cube. A cycle C_{2n} is a partial cube since it is a subdivision $S(C_n)$ $n \geq 3$. It is obvious to see that a subdivision of a *paw* (a cycle C_3 with pendant vertex) is a partial cube if the subdivided C_3 is an even cycle. Moreover, we have the following results:

Proposition 1. *[9] For any $n \geq 1$, $S(K_n)$ is a partial cube.*

Lemma 1. *[7] Let G be a graph and let K be an isometric subgraph of G which is isomorphic to a subdivision of $K_{2,3}$. Then G is not a partial cube.*

This result is very useful to provide the necessary condition of a subdivided connected graph as partial cube. In fact, if such subdivision contains a $sub(K_{2,3})$ as an isometric subgraph, this subdivision is not a partial cube.

Lemma 2. *Let G be a subdivision of a diamond $(K_4 - e)$. G is a partial cube if and only if the chord of the diamond is not subdivided.*

Proof. Let G be a subdivision of the diamond $(K_4 - e)$ induced by vertices w_1, w_2, w_3, w_4 such that w_1w_3 is its chord. The edge w_1w_3 does not contain added vertices, otherwise G is isomorphic to a subdivision of $K_{2,3}$. Hence G is not a partial cube. However, when the vertices w_1 and w_3 are adjacent in G, this latter is isomorphic to a combination of two even isometric cycles C_{n_1} and C_{n_2} and having a common edge w_1w_3. The equivalence classes of the relation θ have exactly two edges except the class that intersects G in w_1w_3. Notice that $w_1w_3\theta e$ and $w_1w_3\theta f$ where e and f are edges of C_{n_1} and C_{n_2} respectively. It is straightforward to see that $e\theta f$. More than, each edge different than e in C_{n_1}(resp. than f in C_{n_2}) is not in relation with f (resp. with e). Then, θ is transitive.

Let W_k be the k-wheel, that is, the graph obtained as a join of the one vertex graph K_1 and all the vertices of the cycle C_k. We denote the central vertex of W_k by u and the remaining vertices by $w_1, ..., w_k$. $W_k(m_1, ..., m_k; n_1, ..., n_k)$ is the graph obtained by subdividing edges of W_k, where m_i is the number of vertices added on the edge w_iw_{i+1}, and n_i the number of vertices added on the inner edge uw_i. $w_{i,1}, ..., w_{i,m_i}$ are the vertices added on edges w_iw_{i+1} and $u_{i,1}, ..., u_{i,n_i}$ the added vertices on edges uw_i for $m_i > 0$ and $n_i > 0$. See figure 1.b.

Theorem 1. *[7] Let $k \geq 3$. Then a subdivided wheel W is a partial cube if and only if W is isomorphic to $W_k(m_1, ..., m_k; n_1, ..., n_k)$, where m_i is odd for $i = 1, ..., k$ and $n_1 = ... = n_k = 0$, or $W = W_3(1, 1, 1; 1, 1, 1)$.*

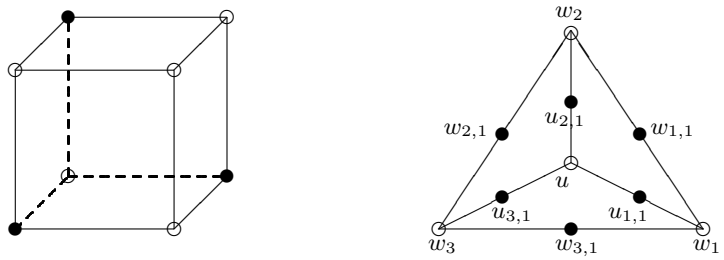


Fig. 1. Isometric embeddings of Q_3^- and $S(K_4)$

In literature, the subdivision of the wheel has been treated as partial cubes and important results were provided. The subdivided wheels result was interesting since it consists in answering in negative a question of Chepoi and Tardif [2] whether partial cubes are precisely bipartite graphs with convex intervals.
An interesting instance of partial cubes mentioned in the last theorem is the graph or $W_3(0,0,0;1,1,1)$. It is defined as the vertex-deleted subgraph $Q_3 - v$ where $v \in Q_3$.

Note that, it is isomorphic to any of $W_3(1,0,0;1,1,0)$, $W_3(0,1,0;0,1,1)$ and $W_3(0,0,1;1,0,1)$. See figure 1.a.

For $k \geq 3$, the k-fan F_k is the graph obtained as the join of a vertex u and a path on k vertices $w_1, ..., w_k$. For each graph G having an isometric subgraph which is isomorphic to $sub(F_k)$ such that $uw_2, ..., uw_{k-1}$ is subdivided, G is not a partial cube. However, we complete this result established in [7] by asserting the following lemma:

Lemma 3. *Each subdivided k-fan F_k having no added vertex on the edge uw_j ($j \in \{2, ..., k\}$) is a partial cube.*

Proof. Let $G = sub(F_k)$ where F_k is a k-fan ($k \geq 3$) induced by the vertices $u, w_1, ..., w_k$. Assume that $uw_2, ..., uw_{k-1} \in E(G)$, let us show that G is a partial cube by induction on k. If $k = 3$, G is a subdivision of a diamond. According to the previous result, G is a partial cube. For $k > 3$, we observe that each subgraph H of G which is isomorphic to a subdivision of l-fan F_l in $sub(F_k)$ is isometric where $l < k$. By the induction hypothesis, H is a partial cube. Moreover, it's straightforward to see that the transitivity is conserved in G, since H is a partial cube. Hence, G is a partial cube.

Theorem 2. *[1] Let K_4 be a complete graph induced by w_1, w_2, w_3, w_4 and let G be a graph obtained from K_4 by joining a vertex u to vertices w_3, w_4. $Sub(G)$*

is a partial cube if and only if the subgraph $H = sub(K_4)$ *is isomorphic to the subdivided wheel* $W_3(m_1, m_2, m_3; 0, 0, 0)$ (m_i *odd* $i \in \{1, 2, 3\}$) *centred in* w_3 (*or* w_4) *and the sum numbers of added vertices on the edges* uw_3 *and* uw_4 *is odd.*

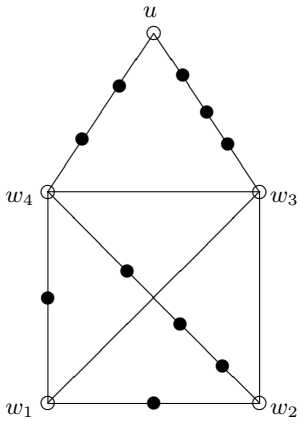


Fig. 2. $sub(G)$ is partial cube if w_3 is a universal vertex in H.

Lemma 4. *Let* G *be a subdivision of* K_n ($n \geq 4$) *where each edge in* K_n *is an isometric path in* G. G *is a partial cube if and only if* G *contains a universal vertex and the other edges of* K_n *have exactly one added vertex or* G *is isomorphic to* $S(K_n)$.

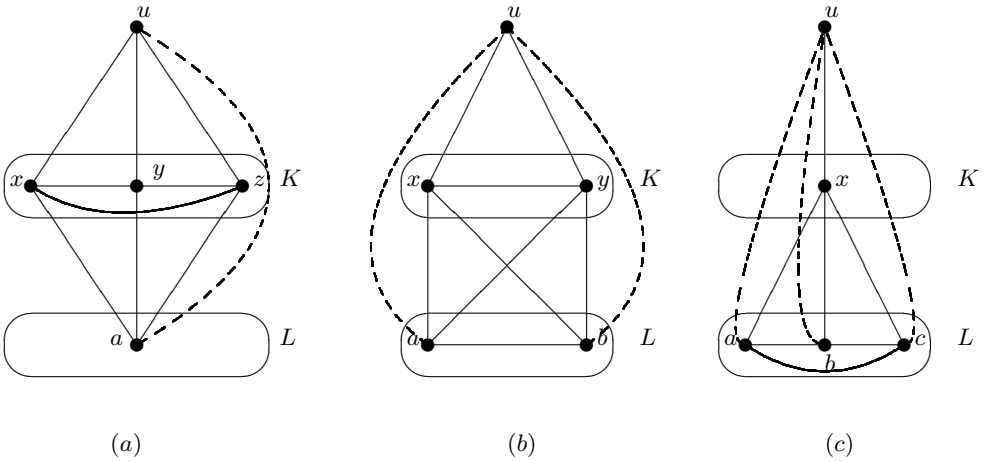
Proof. Let G be a subdivision of the complete graph K_n induced by $w_1, ..., w_k$. Suppose that all the paths $P(w_i, w_j)$ $i \neq j$, $i, j \in \{1, 2, 3, 4\}$ are isometric. Assume that G is a partial cube, and let us show by induction on n that G is either isometric to $S(K_n)$ or G contains a universal vertex. If $n = 4$, G is a 3-wheel W_3. According to the precedent theorem G is precisely either isomorphic to $W_3(1, 1, 1; 1, 1, 1)$ or $W_3(0, 0, 0; 1, 1, 1)$. Assume that $n > 4$, and let G' be the subdivision of the complete graph K_{n-1} induced by $w_1, ..., w_{k-1}$. Since G' is an isometric subgraph of G, G' is a partial cube. By induction hypothesis, G' has a universal vertex w_1 or G' is isomorphic to $S(K_{n-1})$. In the first case, for each distinct index i and j with $i, j \in \{2, ..., k-1\}$ the subgraph H which is a subdivision of a complete graph K_4 induced by w_1, w_k, w_i, w_j is isometric and isomorphic to $W_3(0, 0, 0; 1, 1, 1)$ where $i \neq j$. Then, w_1 is a universal vertex in G. On the other case, the subgraph H is either isomorphic to $S(K_4)$ or contains a universal vertex w_k. Then, G is either $S(K_n)$ or has w_k as universal vertex.

Since $S(K_n)$ is a partial cube, the remaining proof consists of showing that each $sub(K_n)$ having a universal vertex and exactly one added vertex to the

other edges of K_n is a partial cube. In fact, each isometric subgraph isomorphic to $sub(K_{n-1})$ is a partial cube. The transitivity of the relation θ in this subgraph is sealed in G. Consequently, the relation θ is transitive in G.

Proposition 2. *Let G be a subdivision of K_5. G is a partial cube if and only if G is isomorphic to $S(K_5)$ or G contains a universal vertex u and the number of the added vertices to each edge non adjacent to u in K_5 is odd.*

Proof. Necessary condition. Let G be a subdivision of K_5 and $u \in V(K_5)$ such that there exists at least one neighbor x of u where the corresponding path $P(u,x)$ in G is not isometric. We have to note that the other case have been treated above. Let H be the isometric subgraph of G resulting from deleting the inner vertices of all non geodesic paths $P(u,x)$. Observe that H is isomorphic to $sub(K_5 - K_{1,p})$ with $1 \leq p \leq 3$. Let K (resp. L) be the vertices set of K_5 adjacent to u such that the corresponding paths in G is geodesic (resp. non geodesic). It is straightforward to note that the sets K and L are not empty. Let us show that if G is a partial cube then G contains a universal vertex which belongs to K.



(a) (b) (c)

Fig. 3.

Case 1. If H is isomorphic to a subdivision of $K_5 - K_{1,1}$, then $K = \{x, y, z\}$ and $L = \{a\}$(See Figure 3.a). Hence; there exists two vertices in $\{x, y, z\}$ -let assume that it is x and y- such that the subdivided complete graph K_4 induced by u, x, y, a is isometric. This graph is isomorphic to $W_3(0,0,0;1,1,1)$ otherwise if each edge of K_4 has exactly one added vertex, and consequently $P(u,a)$ is isometric.Contradiction. However, if the subdivided wheel $W_3(0,0,0;1,1,1)$ has a universal vertex$u$, the subdivided $K_4 - e$ induced by a, x, y, u is isometric. Since G is partial cube, the vertex x meets the vertex y. Consequently, G contains K_3

and G is not bipartite. Hence, the universal vertex of H belongs to K. *Case* 2. When $p = 2$, the subgraph H is isomorphic to $sub(K_5 - K_{1,2})$. According to the previous lemma the subdivided complete graph K_4 induced by x, y, a, b is the subdivided wheel $W_3(m_1, m_2, m_3; 0, 0, 0)$ having a center reached in a vertex of K, for instance the vertex y. The subdivided complete graph K_4 induced by u, x, y, b is isometric. Consequently, the vertex y is universal in H.
Case 3.
The subgraph H is isomorphic to $sub(K_5 - K_{1,3})$, we set $K = \{x\}$ et $L = \{a, b, c\}$. The subgraph of G which is a subdivision of K_4 induced by the set of K and L is isometric and isomorphic to a subdivided wheel W_3. This one is not isomorphic to $S(K_4)$ otherwise, there exists a subdivided diamond having a subdivided chord. If the subdivided contains a universal vertex, this one is not reached in L otherwise the isometric subdivided diamond induced by a, b, x, u has a non subdivided edge xb. Then G is not bipartite.

Sufficient condition. Now, let us suppose that G is a subdivision of K_5 such that G contains a universal vertex u and each edge w_iw_j in K_5 with w_j and w_j are different than u, has an odd number of added vertices in G. If n is the sum number of added vertices to edges of K_5, then it is clearly that $n \geq 6$. Let us show by induction on n, that G is a partial cube. For $n = 6$, we observe that each edge in K_5 is a geodesic path in $sub(K_5)$. According to the previous lemma, G is a partial cube. Now, let us suppose that $n > 6$. Then, there exists an edge w_1w_2 in K_5 having the added vertices $w_{1,1}, w_{1,2}, ..., w_{1,2k+1}$ where $k \geq 1$. The graph G' obtained from G by removing the vertices $w_{1,2k}, w_{1,2k+1}$ is a partial cube by induction hypothesis. G can be constructed from G' by an expansion over the proper cover $G'_1 = \{w_1, w_{1,1}, ..., w_{1,k}\}$ and $G'_2 = V(G') \setminus G' \cup \{w_1, w_{1,k}\}$. We note that if $k = 1$, we put $G'_1 = \{w_1, w_{1,1}, w_2\}$ and $G'_2 = V(G') \setminus G' \cup \{w_1, w_2\}$. Consequently, G is a partial cube.

4 Open questions

In this paper, we have contributed to characterize partial cubes as subdivisions of a given configuration. From this study, we have established interesting results however more questions remain still asked. For instance, the last result conjectured is very important to verify it since it extends the characterization of $sub(K_5)$ to the general case $sub(K_n)$. Note that this conjecture is true for an order n in $\{4, 5, 6\}$. On the other side, within this paper, we have mentioned some subdivisions which are not partial cubes like $sub(K_{2,3})$. So, It would be interesting to check for the characterization of a partial cube by providing a list of forbidden isometric subgraphs. In [9], Klavzar and Lipovec have studied partial cubes G with the property that removing any edge of G destroys a possibility of its isometric embedding into a hypercube. Small edge-critical partial cubes have been given. The question which comes to minds is among the partial cubes introduced in this articles which ones are edge-critical?

References

1. M. Aïder, S. Gravier, K. Meslem, *Isometric embeddings of subdivided connected graphs in the hypercube*, *Submitted* (2006).
2. V. Chepoi, *d-convexity and isometric subgraphs of Hammming graphs*, Cybernetics **1** (1988) 6–9.
3. D. Djokovič, *Distance preserving subgraphs of the hypercubes*, J.Combin. Theory, Ser **B41** (1973), 263–267.
4. R.L Graham, H.Pollack *On the addressing problem for loop switching*, Bell System Technol. J.50 (1971) 2495–2519.
5. W.Imrich, S. Klavzar, *Product Graphs: structure and recognition*, Wiley, New York,2000.
6. S. Gravier, S. Klavžar, M. Mollard *Isometric embeddings of subdivided wheels in hypercubes*, Discrete Mathematics **263** (2003) 287–293.
7.
8. S. Klavžar, A.Lipovec *Partial cubes as subdivision graphs and as generalized Petersen graphs*, Discrete Math, **263** (2003) 157–165.
9. S. Klavžar, A.Lipovec *Edge-critical isometric subgraphs of hypercubes*, Preprint series, Vol.**39** (2001); 157–165.
10. P.M. Winkler, *Isometric embeddings in Hamming graphs*, Discrete Appl. Math. **7** (1984) 221–225

Résolution d'un système dynamique linéaire avec une commande polyèdrale

M.AIDENE, *B.OUKACHA**, F.KARA

*Département de Mathématiques, Faculté des Sciences
Université Mouloud Mammeri, Tizi-Ouzou, Algérie
oukachabrahim@yahoo.fr

Abstract. Un algorithme à été proposé pour la résolution d'un problème de contrôle optimal d'un système dynamique linéaire avec un temps terminal fixé et une contrainte sur le signal de sortie. La commande agissant sur le système est une fonction vectorielle bornée et continue dans un polyèdre. Après avoir obtenu l'état du système donné par la résolution de l'équation différentielle, le problème initial est transformé en un problème d'optimisation ayant comme seule variable la commande. Ce dernier sera résolu par les techniques de la programmation linéaire, en proposant un algorithme de point intérieur et une procédure qui fait appel à la méthode de Newton.

Mots clés: Optimale Contrôle, Principe du maximum, Support Contrôle.

Classification AMS: 49B10, 49B34, 47N10, 49J15, 49J35, 49Kxx.

1 Introduction

Le problème de contrôle optimale considéré est transformé en un problème de la variable $u(t), t \in T$, qui rentre dans la classe des problèmes de programmation continue ou de programmation semi-infinie. Cette classe de problème constitue un grand intêret pour les chercheurs de par le monde.
En s'appuyant sur la méthode adaptée appliquée à différents problèmes de contrôle optimal [1,2,3,4,5], on a donné le support (base), construit le critère d'optimalité et de suboptimalité. Une itération de l'algorithme a été réalisée.

2 Position du problème

Considérons le problème suivant:

$$J(u) = c'x(t_*) \longrightarrow max, \tag{1}$$

$$\dot{x}(t) = Ax(t) + Bu(t), t \in T, x(t_0) = x_0, \tag{2}$$

$$Hx(t_*) = g, t \in T = [t_0, t_*], \tag{3}$$

$$u(t) \in U,\ U = \{u \in R^r, Du(t) = f, d_* \leq u \leq d^*\}, \tag{4}$$

où:
$x(t) = (x_j(t), j \in J)$ est un n-vecteur représentant l'état du système à l'instant $t \in T, x(t_0)$ la position initiale, $u(t) = (u_j(t), j \in J)$ est un r-vecteur représentant la commande agissant sur le système à l'instant $t \in T$. A,B et$D$ sont des $n \times n$-, $n \times r$-, $k \times r$-matrices données. g et f sont des m- et r-vecteurs donnés. H une $m \times n$-matrice et $rangH = m \leq n$, $c = c(J)$ est un $r-$vecteur.
La solution du système différentiel (2) donnée par la formule de cauchy est égale à:

$$x(t) = F(t)(x_0 + \int_{t_0}^{t_*} F^{-1}(\tau)Bu(\tau)d\tau), t \in T,$$

où F(t) est la solution (résolvante) du système $\dot{F} = AF, F(t_0) = I_n$, I_n représente la matrice identité.
En utilisant cette dernière solution le problème (1)-(4) devient un problème de la seule variable u(t):

$$J(u) = c'F(t_*)x_0 + \int_{t_0}^{t_*} c'(t)u(t)dt \longrightarrow max, \tag{5}$$

$$\int_{t_0}^{t_*} p(t)u(t)dt = g - HF(t_*), \tag{6}$$

$$Du(t) = f, t \in [t_0, t_*],$$

$$d_* \leq u(t) \leq d^*, t \in [t_0, t_*],$$

où

$$p(t) = HF(t_*, t)B, c(t) = c'F(t_*, t)B, F(t_*, t) = F(t_*)F^{-1}(t), t \in T. \tag{7}$$

Définition 1:

Toute commande $u(t)$ vérifiant les contraintes (6)
$\begin{cases} Hx(t_*) = g \\ u \in U = \{u \in R^r, Du = f, d_* \leq u \leq d^*\} \end{cases}$
est dite commande admissible du problème.
Une commande admissible $u^0 = u^0(t)$ est alors dite optimale si $J(u^0) = \max_u J(u)$, et elle est ϵ-optimale si $J(u^0) - J(u^\epsilon) \leq \epsilon$.

3 Support-contrôle

Dans l'ensemble T, choisissons un ensemble fini T_u formé de points isolés $t_j \neq t_*$, et posons $T_u^* = T_u \bigcup t_*$ dont les éléments sont appelés moments du support de la commande. A chaque moment $t \in T_u^*$ on associe l'ensemble $J_u(t) \subset J$ appelé support de la commande au moment t et l'ensemble $Q_u = \{T_u^*; J_u(t), t \in T_u^*\}$ support de la commande si $\det D_u(t) \neq 0$, où $D_u(t) = D(K, J_u(t)), t \in T_u^*$, $|J_u(t)| = k$. De l'ensemble T_u^*, on prend un sous ensemble T_{ux}, $|T_{ux}| \leq m$.
A chaque moment $t \in T_{ux}$, on construit l'ensemble

$$J_{ux}(t) \subset J \backslash J_u(t), \sum_{t \in T_{ux}} |J_{ux}(t)| = m$$

En utilisant cet ensemble, on construit la matrice:

$$P_c = (HF(t_*, t) \times \tilde{B}(I, J_{ux}), t \in T_{ux}),$$

où

$$\tilde{B}(I, J) = B(I, J) - B(I, J_u(t)) D_u^{-1}(t) D(K, J).$$

Définition 2:

L'ensemble $Q_x = \{T_{ux}; J_{ux}(t), t \in T_{ux}\}$ est appelé support des contraintes si $\det P_c \neq 0$. L'ensemble $Q_p = \{J_u(t), t \in T_{ux}; Q_x\}$ est appelé support du problème.
La paire $\{u, Q_p\}$ formée de la commande $u(t)$ et de l'appui Q_p est appelée support contrôle du problème (1)-(4).

Définition 3:

Le support $\{u, Q_p\}$ est dit non dégénéré si l'une des relations suivantes est vérifieé, $\forall \tau \in T_{ux}, j \in J_u(\tau) \bigcup J_{ux}(\tau)$ on a:

1. τ est un point de discontinuité de la fonction $u_j(t), t \in T$
2. au voisinage de τ la composante $u_j(t), t \in T$: $d_{*j} < u_j(t) < d_j^*$, $t \in [\tau - \delta, \tau + \delta], \delta > 0$.

4 Formule d'accroissement de la fonctionnelle

Soit $\{u, Q_p\}$ un support-contrôle de départ, $x(t)$ sa trajectoire correspondante. A l'aide de support Q_p, construisons le vecteur des potentiels y:

$$y' = (c' F(t_*, t) \tilde{B}(I, J_{ux}(t)), t \in T_{ux})' P_c^{-1} \tag{8}$$

et la co-commande (estimateur):

$$\Delta(t) = B' \psi(t) + D' \omega(t), t \in T, \tag{9}$$

où $\psi(t)$ est la solution du système souvant appelé conjugué:

$$\dot{\psi}(t) = -A'\psi, \psi(t_*) = H'y - c \tag{10}$$

et les fonctions $\omega(t), t \in T$ sont égales à:

$$\omega'(t) = (c'F(t_*, t)B(I, J_u(t)), t \in T_{ux})D_u^{-1}. \tag{11}$$

La co-commande $\Delta(t)$ peut être écrite d'une autre manière:
$\Delta(t)$=$\psi'(t)B + \omega'(t)D$
$= (y'H - c')F(t_*, t)B + \omega'(t)D$
$= y'HF(t_*, t)B + \omega'(t)D - c'F(t_*, t)B$
$= y'p(t) + \omega'(t)D - c'(t)$
d'où

$$\Delta(t) = (y', \omega') \begin{pmatrix} \mathrm{p(t)} \\ \mathrm{D} \end{pmatrix} - c'(t), t \in T. \tag{12}$$

Soit $\bar{u}(t) = u(t) + \Delta u(t)$ une autre commande admissible, et
$\bar{x}(t) = x(t) + \Delta x(t), t \in T$ sa trajectoire correspondante.
Déterminons alors l'accroissement de la fonctionnelle:

$$\begin{aligned} \Delta J(u) =& J(\bar{u}) - J(u) \\ &= \int_{t_0}^{t_*} c(t)\Delta u(t)dt \\ &= \int_{t_0}^{t_*} (-\Delta(t) + y'p(t) + \omega' D)\Delta u(t)dt \\ &= -\int_{t_0}^{t_*} \Delta(t)\Delta u(t)dt + \int_{t_0}^{t_*} y'p(t)\Delta u(t)dt + \int_{t_0}^{t_*} \omega'(t)D\Delta u(t)dt. \end{aligned}$$

De l'admissibilité de u et $\bar{u}$, on a:

$$\int_{t_0}^{t_*} y'p(t)\Delta u(t)dt = 0, \int_{t_0}^{t_*} \omega'(t)D\Delta u(t)dt = 0$$

De là,

$$\Delta J(u) = -\int_{t_0}^{t_*} \Delta(t)\Delta u(t)dt \tag{13}$$

5 Calcul de la valeur de suboptimalité

De l'admissibilité de la nouvelle commande $\bar{u}(t) = u(t) + \Delta u(t)$ on a:

$$d_* - u(t) \leq \Delta u(t) \leq d^* - u(t) \tag{14}$$

Le maximum de la fonctionnelle (13) sous les contraintes (14) est atteint pour:

$$\begin{cases} \Delta u_j(t) = d_{*j} - u_j(t) & \text{si } \Delta_j(t) > 0, \\ \Delta u_j(t) = d_j^* - u_j(t) & \text{si } \Delta_j(t) < 0, \\ d_{*j} - u_j(t) \leq \Delta u_j(t) \leq d_j^* - u_j(t), & \text{si } \Delta_j(t) = 0, t \in T, \end{cases}$$

et est égal à:

$$\beta(u,Q_p)=\sum_{j=1}^{r}\int_{T_j^+}\Delta_j(t)(u_j(t)-d_{*j})dt+\sum_{j=1}^{r}\int_{T_j^-}\Delta_j(t)(u_j(t)-d_j^*)dt, \quad (15)$$

appelèe valeur de suboptimalité; où $T_j^+=\{t\in T,\Delta_j(t)>0\}$,
$T_j^-=\{t\in T,\Delta_j(t)<0\}$.
De là $\Delta J(u)=J(\bar{u})-J(u)\leq B(u,Q_p)$.
En vertu de cette inégalité on obtient les critères suivants.

6 Critère d'optimalité

Théorème 1

les relations:

$$\begin{cases}\Delta_j(t)>0 & ,si\ u_j(t)=d_{*j};\\ \Delta_j(t)<0 & ,si\ u_j(t)=d_j^*;\\ \Delta_j(t)=0 & ,si\ d_{*j}<u_j(t)<d_j^*,t\in T,\end{cases} \quad (16)$$

sont suffisantes et dans le cas de la non dégénéréscence elles sont nécessaires pour l'optimalité du support-contrôle $\{u,Q_p\}$.

preuve

condition suffisante:
Si les relations (16) sont verifiées alors de (15) on a $B(u,Q_p)=0$.
Comme $J(\bar{u})-J(u)\leq B(u,Q_p)=0$, pour tout $\bar{u}$, alors $\{u,Q_p\}$ est un support-contrôle optimal.
condition nécessaire:
on va procéder par l'absurde: soit $\{u,Q_p\}$ un support-contrôle optimal non dégénéré et supposons que les relations (16) ne soient pas vérifiées,
c'est dire, il existe $t_1\in T:\Delta_j(t_1)>0$ et $u_j(t_1)>d_{*j}$, ou $\Delta_j(t_1)<0$ et $u_j(t_1)<d_j^*$.
Choisissons un cas par exemple: $\Delta_j(t_1)>0$ et $u_j(t_1)>d_{*j}$, donc de la continuité des fonctions $\Delta_j(t)$ et $u(t),t\in T$, il existe $\epsilon>0$ tel que $\Delta_j(t)>0$ et $u_j(t)>d_{*j}$, $t\in[t_1-\epsilon,t_1+\epsilon]$.
On construit une nouvelle commande $\bar{u}(t)=u(t)+\Delta u(t)$ de la manière suivante:
Sur $T\backslash T_{ux}$, on pose l'accroissement $\Delta u(t)$:

$$\Delta u(t)=\begin{cases}\theta(d_*-u(t)),t\in T(t_1)=[t_1-\epsilon,t_1+\epsilon],\theta>0\\ \Delta u_i^\lambda=(\Delta u_{ij}^\lambda,j\in J_u\cup J_{ux}),i=\overline{1,m},t\in[\tau_i-\lambda,\tau_i+\lambda]\\ 0,t\in T\backslash(\cup_{i=1}^m[\tau_i-\lambda,\tau_i+\lambda]\cup T(t_1)),\end{cases}$$

et de l'admissibilit de $\bar{u}(t),t\in T$, on calcule les autres accroissements $\Delta u(t)$, $t\in T_{ux}$. De là on obtient les équations suivantes:

$$\begin{pmatrix} \int_{t_0}^{t_*} p(t)\Delta u(t)dt \\ D\Delta u(t) \end{pmatrix} = \begin{pmatrix} 0 & \sum_{i=1}^{m} \int_{\tau_i-\lambda}^{\tau_i+\lambda} P_c(t)dt \\ D(K, J_u(t)) & D(K, J_{ux}(t)) \end{pmatrix} \Delta u_i^{\lambda} + \begin{pmatrix} \int_{T(t_1)} p(t)\theta(d_* - u(t))dt \\ D(K, J(t_1))\theta(d_* - u(t)) \end{pmatrix}$$

posons:

$$F(\epsilon, \lambda) = \sum_{i=1}^{m} \int_{\tau_i-\lambda}^{\tau_i+\lambda} P_c(t)dt(\Delta u_{ij}^{\lambda}, j \in J_{ux}(t)) + \int_{t_1-\epsilon}^{t_1+\epsilon} p(t)\theta(d_* - u(t))dt.$$

De là on a:

$$F(0,0) = 0, et \ \frac{\partial F(0,0)}{\partial \lambda} = P_c(t) \begin{pmatrix} \Delta u_1^{\lambda} & . & . & 0 \\ . & \Delta u_2^{\lambda} & . & . \\ . & . & . & . \\ 0 & . & . & \Delta u_m^{\lambda} \end{pmatrix}$$

$$\det \frac{\partial F(0,0)}{\partial \lambda} = \det P_c(t) \prod_{i=1}^{m} \Delta u_i^{\lambda} \neq 0.$$

D'aprés le théorème des fonctions implicites pour $\epsilon > 0$ suffisement petit, il existe une fonction continue unique telle que $F(\epsilon, g(\epsilon)) = 0$. Ce qui nous donne:

$$\sum_{i=1}^{m} \int_{\tau_i-\lambda}^{\tau_i+\lambda} P_c(t)dt(\Delta u_{ij}^{\lambda}, j \in J_{ux}) + \int_{t_1-\epsilon}^{t_1+\epsilon} P(t)\theta(d_* - u(t))dt = 0.$$

De là on trouve les accroissements:
$(\Delta u_{ij}^{\lambda}, j \in J_{ux}) = -\Phi_c^{-1} \int_{t_1-\epsilon}^{t_1+\epsilon} p(t)\theta(d_* - u(t))dt,$
où $\phi_c = (\int_{\tau_i-\lambda}^{\tau_i+\lambda} P_c(t)dt, i = \overline{1, m})$.
Considérons le système:

$$D(K, J_u(t))(\Delta u_{ij}^{\lambda}, j \in J_u(t)) + D(K, J_{ux}(t))(\Delta u_{ij}^{\lambda}, j \in J_{ux}(t)) + D(K, J(t_1))\theta(d_* - u(t)) = 0 \quad (17)$$

De l'équation (17) et comme $D(K, J_u(t))$ est inversible, alors:

$$(\Delta u_{ij}^{\lambda}, j \in J_u(t)) = -D^{-1}(K, J_{ux}(t))[D(K, J_{ux}(t))(\Delta u_{ij}^{\lambda}, j \in J_{ux}) + D(K, J(t_1))\theta(d_* - u(t))]$$

Donc pour $\epsilon > 0$ suffisamment petit $\bar{u}(t)$ est admissible.
De la formule d'accroissement de la fonctionnelle (13), on deduit:

$$\begin{aligned}\Delta J(u) &= J(\bar{u}) - J(u) \\ &= -\int_{t_1-\epsilon}^{t_1+\epsilon} \Delta(t)\theta(d_* - u(t))dt - \sum_{i=1}^{m} \int_{\tau_i-\lambda}^{\tau_i+\lambda} \Delta(t)\Delta u_i^{\lambda} dt \\ &= -\theta \int_{t_1-\epsilon}^{t_1+\epsilon} \Delta(t)(d_* - u(t))dt + o(\epsilon) > 0.\end{aligned}$$

Donc pour $\epsilon > 0$ suffisamment petit on a: $J(\bar{u}) - J(u) > 0$, c'est à dire: $J(\bar{u}) > J(u)$, ceci contredit l'optimalité de la commande $u(t), t \in T$, CQFD.

7 Critère du ϵ-minimum

Soit un nombre $\epsilon > 0$ donné. Pour que la commande admissible $u = (u(t), t \in T)$ soit ϵ-optimale, il est nécessaire et suffisant de trouver un tel appui Q_p, tel que le long de la commande appui $\{u, Q_p\}$ et des solutions $x(t)$, $\psi(t)$, $t \in T$ des systèmes direct et conjugué, l'hamiltonien $H(x, \psi, u) = \psi'(Ax + Bu)$ atteint la valeur ϵ-minimale:

$$H(x(t), \psi(t), u(t)) = \min_{u \in U} H(x(t), \psi(t), u) + \epsilon, t \in T,$$

$$\int_{t_0}^{t_*} \epsilon(t) \leq \epsilon.$$

8 Méthode de résolution

Soit $\{u, Q_p\}$ un support contrôle de départ, pour lequel $\beta(u, Q_p) > \epsilon$.
L'itération de l'algorithme $\{u, Q_p\} \rightarrow \{\bar{u}, \bar{Q}_p\}$ est constituée de trois procédures:

1. changement de commande,
2. changement de support,
3. procédure finale.

8.1 Changement de commande

Soit $\{u(t), Q_p\}$ un support contrôle non dégénéré de départ et
$\bar{u}(t) = u(t) + \theta\Delta u(t)$, $t \in T$, une autre commande admissible, où $\Delta u(t)$ est la direction de changement de la commande et θ le pas maximal le long de cette direction. Le pas θ et la direction seront cherchés comme solution du problème suivant:

$$\Delta J(u) = -\theta \int_{t_0}^{t_*} \Delta(t)\Delta u(t)dt \to \max_{\theta,\Delta u(t)},$$

$$\int_{t_0}^{t_*} p(t)\Delta u(t)dt = 0, \tag{18}$$

$$D(K,J)\Delta u(t) = 0,$$

$$d_* - u(t) \le \theta\Delta u(t) \le d^* - u(t), t \in T.$$

Choisissons les nombres $\alpha > 0$, $h > 0$ (paramètres de la méthode) et construisons les ensembles suivants:
$T_{0j} = \{t \in T : |\Delta_j(t)| \le \alpha\}, j \in J$; $J_z(t) = \{j \in J; t \in T_{0j}\}$;
$T_z = \{t \in T; |J_z(t)| > k\}$, $T_* = T \backslash T_z$.
Subdivisons l'ensemble T_z en intervalles $[\tau_i, \tau^i], i = \overline{1,M}$ et l'ensemble T_* en intervalle $[\tau_i, \tau^i], i = \overline{M+1,s}$ tels que:

$$0 < \tau^i - \tau_i \le h, i = \overline{1,M};]\tau_i, \tau^i[\cup]\tau_j, \tau^j[= \emptyset, i \ne j, \cup_{i=N+1}^{s} [\tau_i, \tau^i] = T_*$$

$u(t) = u_i = (u_{ij}, j \in J) = \text{const}$, $J_u(t) = J_{u_i} = \text{const}$, $t \in]\tau_i, \tau^i[, i = \overline{1,s}$.
Faisons un changement de variable:
$l_i = (l_{ij}, j \in J) = \theta\Delta u_i = \theta(\Delta u_{ij}, j \in J), i = \overline{1,s}$.
Calculons les qauntités suivantes:

$$g_i = -\int_{t_0}^{t_*} \Delta(t)dt, i = \overline{1,s}\ q_i = \int_{t_0}^{t_*} p(t)dt = \int_{t_0}^{t_*} HF(t_*,t)Bdt, i = \overline{1,s}$$
$$\tilde{d}_{*i} = \tilde{d}_{*ij} = (d_{*j} - u_{ij}, j \in J); \tilde{d}_i^* = \tilde{d}_{ij}^* = (d_j^* - u_{ij}, j \in J), i = \overline{1,s}$$

En utilisant ces dernières quantités, le problème (18) devient:

$$\sum_{i=1}^{s} \acute{g}_i l_i \to max,$$

$$\sum_{i=1}^{s} q_i l_i = 0, \tag{19}$$

$$Dl_i = 0, \tilde{d}_{*i} \le l_i \le \tilde{d}_i^*, i = \overline{1,s}.$$

Ce problème est un problème de programmation linéaire qu'on va résoudre par la méthode adaptée en prenant comme plan de départ $(l_i = 0; i = \overline{1,s})$ et le support $U_u = \{J_{u_i}, i = \overline{1,s}; J_{ux_i}, i \in S_{ux}\}$, $J_{ux_i} = \emptyset, S_{ux} = \emptyset$.
Après un certain nombre d'itérations on obtient la solution :
$(\bar{l}_i, i = \overline{1,s}), \bar{u}_u = \{\bar{J}_{u_i}, i = \overline{1,s}; \bar{J}_{ux_i}, i \in \bar{S}_{ux}\}$. La nouvelle commande sera alors:

$$\bar{u}(t) = u(t) + \bar{l}_i, t \in]\tau_i, \tau^i], i = \overline{1,s}.$$

La commande $\bar{u}$ ainsi construite vérifie l'inégalité:$J(\bar{u}) \geq J(u)$.
Si $\beta(\bar{u}, Q_p) \leq \epsilon$, alors $\bar{u}$ est ϵ- optimale, si non on passe à une nouvelle itération avec $\bar{u}$ et les paramètres $\bar{\alpha} < \alpha$ et $\bar{h} < h$, ou bien à la procédure de changement de support.

8.2 Changement de support

Soit $\bar{u}, Q_p$ un support contrôle avec $\beta(\bar{u}, Q_p) > \epsilon$. En utilisant Q_p, on calcule la trajectoire $\psi(t), t \in T$, solution du système conjugué.
Avec cette dernière on construit la quasi commande $\omega(t), t \in T$:

$$\omega_j(t) = \begin{cases} d_{*j}, \ si\ \Delta_j(t) > 0, \\ d_j^*, \ si\ \Delta_j(t) < 0, \\ \in U, \ si\ \Delta_j(t) = 0, t \in T, \end{cases}$$

et sa quasi trajectoire correspondante $\chi = (\chi(t), t \in T)$, solution du système
$\dot{\chi} = A\chi + B\omega$, $\chi(0) = \chi_0$.
si $H\chi(t_*) = g$, alors la quasi commande $\omega(t)$ est optimale.
si $H\chi(t_*) \neq g$, construisons alors le vecteur:

$$\lambda(T_{ux}) = R^{-1} \begin{pmatrix} H\chi(t_*) - g \\ 0 \end{pmatrix},$$

où $R = \begin{pmatrix} 0 & \int\limits_{T_{ux}} P_c(t)dt \\ D_u & D_{ux} \end{pmatrix}$, $detR \neq 0$ car $detP_c \neq 0$ et $detD_u \neq 0$.

si $\lambda(T_{ux}) = 0$, alors la quasi-commande ω construite par l'appui Q_p est optimale.
si non calculons $\|\lambda(T_{ux})\|$; deux cas peuvent se présenter:

1. si $\|\lambda(T_{ux})\| > \mu$, où $\mu > 0$ paramètre de la méthode alors on change l'appui Q_p en $\bar{Q}_p$ par la méthode duale.
2. si $\|\lambda(T_{ux})\| \leq \mu$, on passe à la procédure finale.

8.3 Procédure finale

Supposons que $\|H\chi(t_*) - g\| \leq \mu$.
Consrtuisons les ensembles:
$J_0(t) = \{j \in J \backslash J_u(t) : \Delta_j(t) = 0\}, t \in T$ et $T_0 = \{t \in T : J_0(t) \neq \emptyset\}$.
Supposons que T_0 est constitué de points isolés.
Considérons le cas où $T_0 = m$, $J_0(t) = 1$, $t \in T_0$, $detP_c \neq 0$, avec:

$$P_c = (HF(t_*, t)\tilde{B}(I, j), j \in J_0(t), t \in T_0)$$

l'ensemble $Q_c = \{T_{ux}, J_u(t) \bigcup J_{ux}(t), t \in T_{ux}\}$ où $T_{ux} = T_0$, $J_{ux}(t) = J_0(t), t \in T$ est un appui de problème (1)-(4).
Soit $t_i (t_{i-1} < t_i), i = \overline{1, m}$ les points de T_{ux}, donc $J_u(t) = J_{u_i} = const$, $\omega(t) = \omega_i = const$, $t \in]t_{i-1}, t_i[, i = \overline{1, m+1}$, $t_{m+1} = t_*$. Supposons que $\{\omega, Q_c\}$ est non dégénérée, c'est à dire $\omega_{i-1} \neq \omega_i, i = \overline{1, m+1}$, par conséquent $\omega_{i-1}(j_i) \neq \omega_i(j_i), i = \overline{1, m}$, où $\omega_i = (\omega_i(j), j \in J)$; $j_i = J_{ux}(t_i)$. L'appui optimal cherché est alors:
$Q_c^* = \{T_{ux}^*; J_u(t) \bigcup J_{ux}(t), t \in T_{ux}^*\}$
$T_{ux}^* = \{t_i^*, i = \overline{1, m}\}$,$J_u(t_i^*) = J_{u_i}$, $J_{ux}(t_i^*) = j_i$.
Notre but est de trouver les m inconnues t_i^*, vérifiant les m équations:

$$H\chi(T_{ux}^*, t_*) = g, \tag{20}$$

où $\chi(T_{ux}^*, t_*), t \in T$ est la trajectoire correspondante à la pseudo- commande $\omega(T_{ux}^*, t) = \omega_i, t \in]t_{i-1}^*, t_i^i], i = \overline{1, m+1}, t_{m+1}^* = t_*$.
Comme:

$$det\frac{\partial H\chi(T_{ux}, t_*)}{\partial T_{ux}} = \prod_{i=1}^{m}(\omega_{i-1}(j_i) - \omega_i(j_i))detP_c \neq 0,$$

alors on résout le système (20) par la méthode de Newton le système (20) en prenant comme approximation initiale $t_i^* = t_i, i = \overline{1, m}$. pour un $\mu > 0$ suffisament petit.
De là on obtient la solution optimale de (20) $t_i^0, i = 1, ..., m$ avec laquelle on calcule la commande optimale $u^0(t) = \omega(T_{ux}^0, t), t \in T$.

9 Conclusion

L'objectif de cet article a été de résoudre un système dynamique linéaire avec une commande polyèdrale. Tout d'abord, on s'est intéressé au critére d'optimalité, par la suite on a résolu ce problème avec la méthode adaptée du simplexe trois procédures: Changement de commande, changement de support et la procédure finale, qui est essentielle dans la résolution de ce type de problèmes, elle consiste en la construction d'un système d'équations qu'on a résolu par la méthode de Newton et qui nous donnait directement la solution optimale. Pour terminer, nous estimons qu'il est intéressant de compléter ce travail en étudiant le problème dans le cas d'une fonctionnelle min-max.

References

1. M. AIDENE. Algorithme de résolution d'un problème min-max en control optimal. Exposés de l'Académie des Sciences de Bélarus, T.30, 1, (1986), Pages 24-27.
2. M. AIDENE, I.L. Vorob'ev, and B.Oukacha. Algorithm for the solving a linear optimal control problem with minimax performance index. Computational Mathematics and Mathematical physics. Vol 45, No 10, 2005,PP 1691-1700.
3. R. Gabasov. Adaptive method of linear programming. Preprint of the University of Karlsruhe. Institue of Statistics and Mathematics. Karsruhe, Germany(1993).
4. E.A. Kostina-O.I. Kostuokova. Algorithme de résolution d'un problème convexe de programmation quadratique avec des contraintes linèaires et non linèaires. Revue de mathématique, T.42, N 7, Minsk,(2001), P.1012-1026.
5. L.S. Pontriaguin. Mathematical Theory of optimal processus, Interscience. New York, (1962).

Indexation de vidéo couleur non compressée par le contenu à l'aide de la structure d'arbre R

Karim Abdelkader Henni et Lynda Zaoui

Département Informatique, Faculté des sciences, Université Abdelhamid Ibn Badis Mostaganem
Laboratoire Signaux, Données, Systémes, Université des Sciences et de la Technologie Mohamed Boudiaf, B.P.1505 EL M'NAOUAR-ORAN (ALGERIE)

Henni.2005@gmail.com
Zaoui _lynda@yahoo.fr

Résumé. Depuis le début des années 1990, l'indexation et la recherche par le contenu sont devenues un pôle très actif de la recherche; elle est devevue indipensable aux bases de vidéos afin de répondre aux besoins de plusieurs secteurs comme la télésurveillance, la télévision...etc

Cet article décrit notre approche de recherche et d'indexation de vidéos par le contenu, en suivant deux principaux points :

Dans un premier point nous proposons d'appliquer une structure d'indexation appelée "arbre R" pour indexer une vidéo en procédant à son découpage en images individuelles, puis à sa segmentations temporelle et enfin nous sélectionnons les images clefs, l'ensemble de ces images formant le résumé vidéo. Ce premier point correspond à la phase off-line du prototype.

Dans un deuxième point, l'utilisateur a la possibilité d'introduire au système une image requête, et sur la base de calcul de distance de similarité le système renvoie des résultats. Ce deuxième point correspond à la phase on-line du prototype.

Mots clés. Indexation de vidéos par le contenu, arbre R, distance de similarité, segmentation temporelle, image clef, plan, résumé vidéo.

1 Introduction

Avec les progrès de l'informatique et des télécommunications (Internet ADSL, réseaux mobiles,…etc.), les bases de vidéos sont apparues. Ces bases sont amenées à stocker des millions de vidéos, ce qui a engendré de nouveaux besoins dans plusieurs domaines dont nous citons par exemple :

Le secteur de la télévision s'intéresse à accélérer la recherche dans les archives de télévision.
Les téléspectateurs ont besoin d'avoir une idée générale sur la vidéo sans devoir la visualiser entièrement (ç.à.d avoir un résumé de la vidéo).

Dans le domaine de la télésurveillance les utilisateurs ont besoin de détecter et de faire le suivi des objets dans des séquences vidéos filmées.
Le grand public désire retrouver un plan spécifique contenant une image X d'un film ou d'un journal télévisé.

Quelle que soit la base de données sur la quelle l'utilisateur travaille, ce dernier a toujours recours à une interrogation pour avoir une information précise ; la recherche d'une vidéo dans une base de plusieurs milliers de vidéos nécessite un temps considérable ; pour accélérer le processus, le concept d'*indexation* est alors employé.

Au début l'indexation consistait à donner une description textuelle de la vidéo. Ce type de méthodes (basées sur des annotations textuelles) présente des inconvénients. En effet, la description textuelle demande une intervention humaine ; et celle-ci a deux limites principales : le temps et le jugement; sans parler des limitations liées à la langue (ou au jargon) utilisée pour l'indexation.
D'une part, présenter le contenu d'une vidéo demande beaucoup de temps ("voiture bleue qui suit une voiture rouge" par exemple). Pour des ensembles d'une centaine de vidéos cela est encore imaginable mais actuellement les volumes étudiés sont plutôt de l'ordre du millier voir de la centaine de milliers de vidéos. Une intervention humaine pour décrire de tels ensembles semble donc impossible.

Par conséquent, et ne pouvant pas se baser entièrement sur une description textuelle, la recherche de vidéos tente aujourd'hui d'extraire les informations directement des vidéos et d'une manière automatique. Ce type de travaux est couvert par le terme : "*indexation de vidéos basée sur le contenu*". Dans ce type de méthodes (méthodes basées sur le contenu visuel) deux étapes sont importantes : une première étape est la segmentation en groupes d'images appelés "plans". Les plans sont des suites d'images filmées sans interruption. A la fin de chaque plan il existe une coupure. Par l'analyse des images consécutives il est possible de détecter automatiquement un changement de plan. La deuxième étape est d'associer des concepts aux plans détectés, les plans sont souvent représentés par des images clefs qui peuvent être analysés pour obtenir une description du contenu [1].
Il existe deux approches pour la détection des changements des plans : détection sur les vidéos compressée et détection sur les vidéos non compressées.
La détection des plans pour la vidéo non compressée est basée sur l'application de quelques mesures capables d'évaluer les changements entre images successives.
La détection des plans pour la vidéo compressée est basée sur la comparaison de certains coefficients spécifiques au codage MPEG.

2 Structure de l'arbre R

L'arbre R (noté AR) est une structure hiérarchique, dérivée de l'arbre B et utilisée pour indexer les objets spatiaux ou géométriques. Les objets spatiaux sont représentés par des rectangles englobant minimum (noté REM) sur une image [2], ceci est intéressant ; en effet on arrive à décrire des objets très complexes avec un

nombre limité d'octets[1], puisque à chaque objet de cet espace on associe uniquement les coordonnées du REM et une description de l'objet [3].

2.1 Caractéristiques des arbres R

L'arbre R, comme l'arbre B, est un arbre équilibré de degré (ou ordre) *m* et présente les caractéristiques suivantes [2], [4], [5] :

Un noeud interne d'arbre R permet d'adresser les noeuds du niveau suivant de l'arborescence.

Un noeud feuille référence un ensemble d'objets spatiaux.

Un noeud (interne ou feuille) est associé à une partie rectangulaire de l'espace de sorte que :

- Le rectangle associé à un noeud feuille est le rectangle englobant minimum des *rem* des objets spatiaux référencés par le noeud feuille.
- Le rectangle associé à un noeud intermédiaire est le rectangle englobant minimum des rectangles englobants minimums associés à chacun de ces fils.
- La racine de l'arborescence est associée au rectangle englobant minimum de l'espace tout entier.

Chaque noeud contient des entrées qui sont des couples (*rem, oid*) tels que :

- Pour les noeuds internes : rem correspond au rectangle englobant minimum associé à un noeud fils, dont l'adresse est *oid*.
- Pour les noeuds feuilles : rem est le rectangle englobant minimum d'un objet spatial dont l'adresse est *oid*.

Chaque nœud, à l'exception de la racine, contient un nombre d'entrées compris entre *m* et *M*, où *m* est le nombre minimal d'entrées par noeud et *M* est le nombre maximal d'entrées, tels que $0 \leq m \leq [M/2]$; $[M/2]$ étant égale à la partie entière de $(M/2)$.

La racine de l'arbre R possède au moins deux fils sinon c'est une feuille.

Toutes les feuilles apparaissent au même niveau.

A chaque objet est associé une entrée composée du REM et de sa description, celle-ci peut se faire à l'aide de mots clés.

A chaque rectangle de l'image on associe une clé, c'est-à-dire une valeur entière permettant de définir sa position et ses dimensions. Ainsi le parcours dans l'arbre se fera par simple comparaison des clés.

[1] Un rectangle à deux dimensions peur être représenté par 4 nombres de 4 octets, chacun. Si un pointeur prend 4 octets, chaque entrée demande 20 octets.

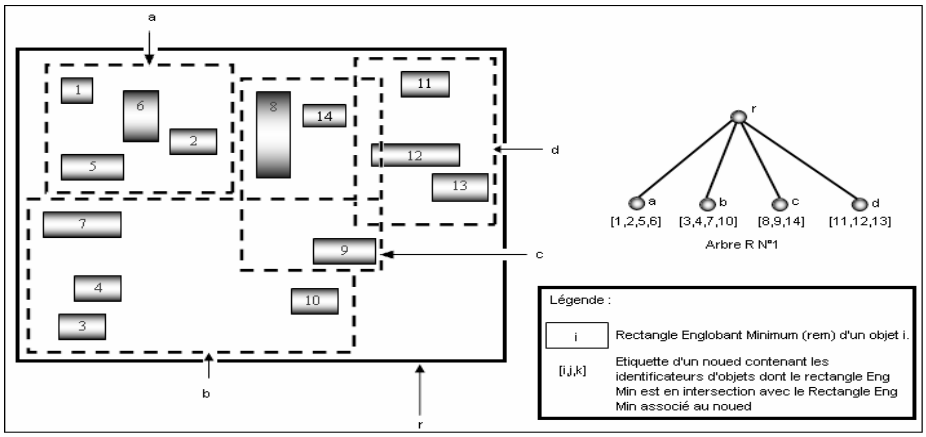


Fig. 1. Un arbre R.

2.2 Distance de R-Similarité

M. Rukoz, M. Manouvrier , G. Jomier dans [6] ont défini une distance générale Δ entre images basées sur les arbres quaternaires, définie par :

$$\Delta(i,j) = \frac{\sum_k c_k \, \delta_k(i,j)}{\sum_k c_k} \qquad (1)$$

– k est l'identificateur d'un nœud pris parmi l'union des identificateurs de nœuds apparaissant dans les arbres quaternaires des images i et j.
– $\delta_k(i, j)$ distances entre nœuds d'arbres quaternaires.
– c_k est un coefficient positif représentant le poids du nœud k dans le calcul de la distance Δ. Le choix de chaque poids c_k dépend des besoins de l'utilisateur, c'est-à-dire de l'importance que l'utilisateur souhaite donner à certains quadrants de l'image par rapport à d'autres dans le calcul de la distance Δ.

Remarque 1. $\Delta(i, j) = 0$ signifie que "tous les nœuds homologues k des arbres quaternaires i et j, de poids c_k non nul, ont une distance δ_k nulle ($\forall$k tel que $c_k \neq 0$: $\delta_k(i, j) = 0$)" et $\Delta(i, j) = 1$ signifie que "tous les nœuds homologues k des arbres quaternaires i et j, de poids c_k non nul, ont une distance δ_k égale à 1 ($\forall$ k tel que $c_k \neq 0 : \delta_k(i, j) = 1$)".

En faisant référence aux arbres quaternaires, les arbres R ont aussi des nœuds, ces noeuds ne correspondent pas aux quadrants retrouvés par un découpage quaternaire mais plutôt aux rectangles englobants minimums des objets spatiaux, nous pouvons définir une distance basée sur les arbres R à partir de la distance Δ.

Nous supposerons que tous les REM ont le même poids dans le calcul de la distance $c_k = 1 \ \forall$ k et donc :

$$\Delta (i, j) = \frac{\sum_k \delta_k(i, j)}{\sum_k c_k} \quad (\mathbf{2})$$

Comme les noeuds d'arbre R ne contiennent pas des informations sur les sous arbres dont ils sont racines ; dans notre cas la distance δ_k ne peut prendre que deux valeurs : 0 ou 1. Si les deux régions à comparer sont identique alors $\delta_k = 0$ sinon $\delta_k = 1$.
Dans $\sum_k \delta_k(i, j)$ on aura que des REM différentes.
$\sum_k c_k$ nous donnera en fait le nombre de REM pris parmi l'union des REM qui apparaissent dans les deux images *i* et *j*.
Notre distance est appelée "*distance de R-similarité*" ; R $(i, j) \in [0, 1]$ et est définie par l'équation suivante :

$$R (i, j) = \frac{| S (i, j) |}{| U (i, j) |} \quad (\mathbf{3})$$

Avec | S (i, j) | : nombre de nœuds et de rem différents entre les images *i* et *j*.
| U (i, j) | : nombre total (sans doublon) des identificateurs des nœuds et des rem apparaissant dans les arbres R des images *i* et *j*.

3 Notre approche

Notre approche d'indexation et de recherche de vidéos par le contenu comporte cinq étapes incontournables décrites dans ce qui suit :

3.1 Découpage en images

Il s'agit dans cette étape de restituer les images individuelles qui ont formé la vidéo grâce aux algorithmes et aux techniques de programmation ainsi que des logiciels de traitement de vidéo et de capture d'écran existant sur le marché comme Virtual dub, OSS video decompiler… ; ces images nous permettront par la suite de définir des plans d'images en utilisant des distances de similarité.

3.2 Segmentation spatiale et construction de l'arbre R

La segmentation d'images constitue un élément clé entre traitement et analyse de l'image, elle permet de passer du support de l'image à une liste d'objets décrits en terme de régions [7].

Dans cette étape il s'agit de segmenter chaque image individuellement (qui est le résultat de l'étape précédente) en utilisant un des algorithmes de segmentation existants.

Cette étape sert à construire l'arbre R pour chaque image segmentée spatialement, et par conséquent à calculer les distances de R-Similarité entre les images, elle est transparente à l'utilisateur et elle a une utilité pour la segmentation temporelle.

3.3 Segmentation temporelle

La segmentation temporelle est une étape très importante pour les approches d'indexation et de recherche de vidéos par le contenu.

La majorité des approches utilisent des critères de bas niveau comme la couleur, la texture ou le mouvement [8], [9] pour segmenter la vidéo en plusieurs unités de base appelées "*plans*", quatre catégories de méthodes existent détaillées ci dessous :

> Les méthodes basées sur les différences pixel à pixel détectent un changement de plan en calculant une différence entre les pixels de l'image à l'instant *t* et ceux de l'image à l'instant *t+1*. Si le nombre de pixels différents est supérieur à un seuil, alors on considère que l'on est en présence d'un "*cut*".
>
> Les méthodes à base d'histogramme comparent deux images successives en s'appuyant sur leurs histogrammes respectifs. Une différence des deux histogrammes est calculée et comparée à un seuil. Un changement de plan est détecté si la différence obtenue est supérieure au seuil. Du fait de l'utilisation d'histogrammes, il est possible de ne pas détecter un *cut* si les deux images concernées ont un histogramme similaire mais un contenu différent.
>
> Les méthodes basées sur une estimation du mouvement utilisent l'information de mouvement comme critère principal pour la détection des changements de plan. Les mouvements sont estimés pour chaque pixel d'une image obtenue à l'instant *t*, et sont comparés avec ceux de l'image correspondant à l'instant *t+1*. Un nombre trop important de mouvements incohérents entre les deux images successives implique alors la détection d'un changement de plan.
>
> Les méthodes basées sur les blocs sont des méthodes intermédiaires entre les méthodes basées pixels (locales) et les méthodes basées histogrammes (globales).

3.3.1 Notre algorithme

Le plan d'images est l'unité technique sur laquelle se base notre approche d'indexation et de recherche de vidéo par le contenu, nous proposons dans ce qui suit un algorithme qui permet de segmenter une vidéo en plans à l'aide des arbres R ; l'algorithme est décrit dans ce qui suit :

1- Obtenir l'ensemble F des images individuelles avec l'étape 1.
2- Construction de l'arbre R pour chaque image de l'ensemble F avec l'étape2.
3- Le 1er plan de la vidéo contient uniquement la 1^{ere} image de l'ensemble F.
4- On calcule la distance de R similarité entre l'image suivante et la 1^{ere} image du plan courant.
5- Si la distance est inférieure au seuil (ce dernier est défini de manière empirique sur un jeu de tests) alors cette image fait partie de ce plan et.
5.1- Tant qu'on a pas atteint la fin de l'ensemble F, on passe à l'image suivante de la séquence et.
5.2- Revenir au 5.
6- Sinon l'image courante génère un nouveau plan et aller à l'étape 4.
7- Dés qu'on atteint la fin de l'ensemble F, on s'arrête.

3.4 Sélection de l'image clef

Après avoir déterminé les plans dans l'étape précédente, nous devons extraire dans cette étape les caractéristiques visuelles de chaque plan. Ces caractéristiques sont définies dans une ou plusieurs images appelées "*images clefs*" (appelées parfois images représentatives, ou images caractéristiques).

Différentes approches ont été proposées jusqu'ici pouvant être classées en trois catégories :

Approches se basant sur les différences visuelles entre images : ces approches utilisent donc les caractéristiques brutes obtenues sur les images (couleur, texture, mouvement).
Approches se basant sur les configurations du mouvement : ces approches sélectionnent les images représentatives en fonction de l'intensité ou des variations du mouvement.
Une approche différente consiste à créer une mosaïque du plan. Une mosaïque est construite à partir des images du plan et fournit une représentation complète et compacte de la vidéo. Cette approche est beaucoup plus rare car les mosaïques sont difficiles à construire dans le cas général [10].

L'image clef dans notre approche est la première image de chaque plan car le plan contient les images similaires à cette image ; si la distance de similarité est supérieure au seuil nous concluons qu'elle est différente de la première image et forme à son tour un autre plan.

Les images clefs sont les images les plus riches en information par rapport aux autres images. L'ensemble de ces images forme ce que l'on appelle "*résumé vidéo*".

En fait ces quatre étapes correspondent à la phase off-line du prototype, la dernière étape correspond à la phase on-line.

3.5 La recherche dans les vidéos

Dans les bases d'images, les recherches consistent à savoir si l'image requête (l'image introduite par l'utilisateur dans la phase on-line) est stockée dans cette base d'images ou bien à extraire de cette base toutes les images similaires à l'image requête au sens d'une distance de similarité.

Les bases de vidéos ne sont aujourd'hui qu'à leur début, nous pouvons par conséquent imaginer quelques définitions de recherche dans les vidéos selon les domaines d'application de l'indexation et des besoins des utilisateurs, par exemple la recherche dans les vidéos consiste en :

La recherche de l'exacte réplique de la vidéo introduite par l'utilisateur dans la base de vidéos.
Rechercher des vidéos proches de la vidéo requête.
Détection des objets particuliers (visages, véhicules, bâtiments,….) dans les bases de vidéo.
Rechercher dans la base une vidéo contenant l'exacte réplique d'une image requête ou bien un ensemble d'images similaires à l'image requête.

Nous avons tracé en bas un schéma explicatif montrant les deux phases de notre système d'indexation et de recherche de vidéo par le contenu pour les deux derniers types de recherche énoncés plus haut.

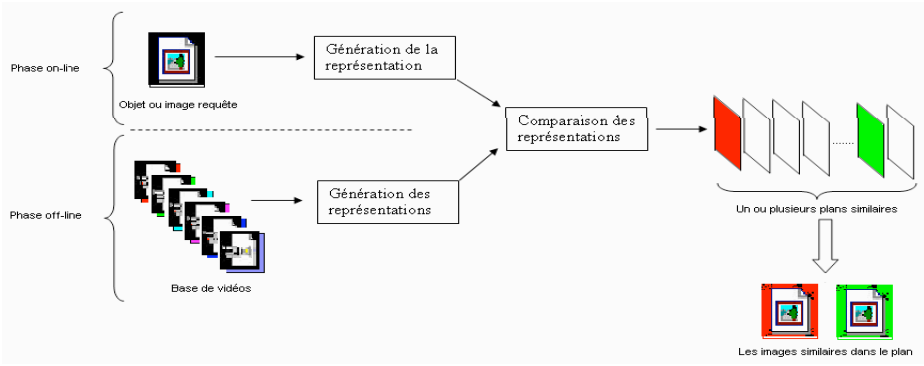


Fig. 2. Principe de fonctionnement d'un système d'indexation de vidéos par le contenu.

4 Résultats d'expérimentation

Nous avons évalué notre approche sur une séquence vidéo couleur non compressée (d'une forte variation du contenu) au format AVI ayant les caractéristiques suivantes :

Taille (size) : 128 x 128
Longueur (lenght) : 0 :07 :000 (7 secondes)
Images (frames) : 105
FPS (frames per seconds) : 15.00

La 1ére étape consiste à découper la vidéo en images individuelles, les résultats sont illustrés par la figure suivante :

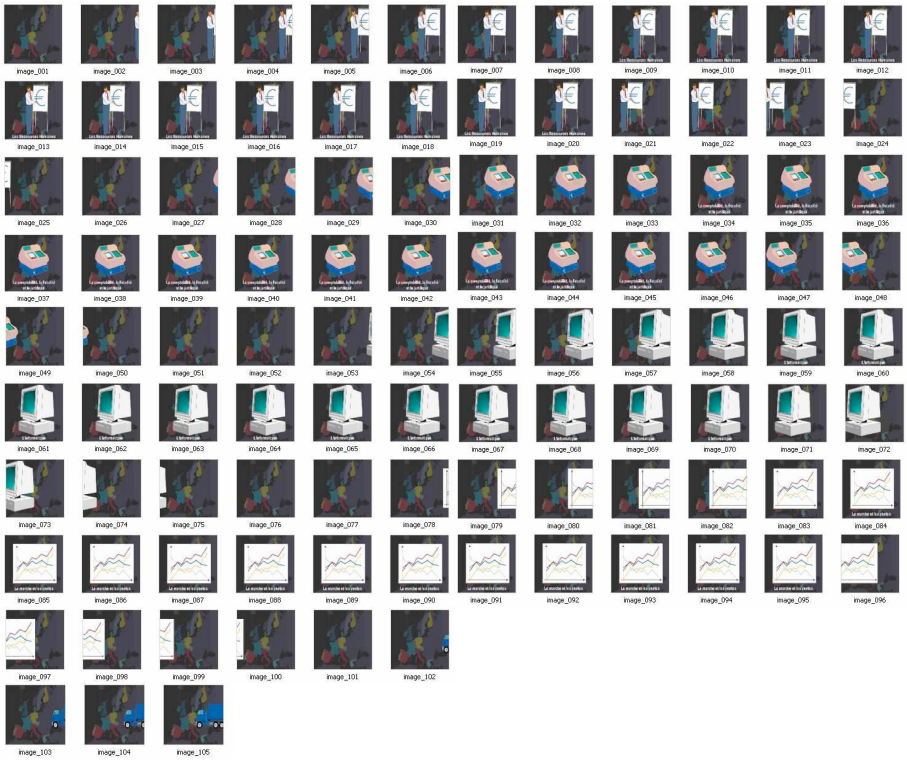


Fig. 3. Images individuelles de la vidéo de l'exemple.

La 2eme étape de notre approche n'est pas illustrée par le prototype car c'est une étape de codage, elle est transparente à l'utilisateur, dans cette étape chaque image est segmentée spatialement et stockée sur disque sous forme d'arbre R.
La 3eme étape consiste à segmenter la vidéo en plans en se basant sur les arbres R des images, les résultats sont les suivants :

Le plan 0 englobe les images : 1,2,…,20 (20 images).
Le plan 1 englobe les images : 21 (1 image).
Le plan 2 englobe les images : 22,23,…,29 (8 images).
Le plan 3 englobe les images : 30,31,…,33 (4 images).
Le plan 4 englobe les images : 34,35,…,52 (19 images).
Le plan 5 englobe les images : 53,54,…,70 (18 images).
Le plan 6 englobe les images : 71,72,…,77 (7 images).
Le plan 7 englobe les images : 78,79,…,82 (5 images).
Le plan 8 englobe les images : 83,84,…,95 (13 images).
Le plan 9 englobe les images : 96,97,…,101 (6 images).
Le plan 10 englobe les images : 102,103,…,105 (4 images).

La 4eme étape consiste à extraire les images clefs des plans, ce qui donne un résumé de la vidéo, les résultats sont illustrés par la figure suivante :

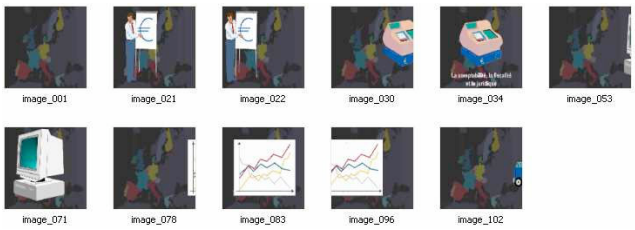


Fig. 4. Résumé vidéo.

Ces étapes correspondent à la phase off-line du prototype, l'utilisateur a la possibilité d'interroger la base lors de la phase on-line, nous avons introduit dans notre prototype deux images requêtes, une qui fait partie de la vidéo et l'autre qui n'en ne fait pas, les deux images requêtes sont illustrées par la figure suivante :

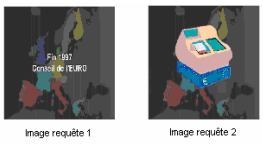


Fig. 5. Deux images requêtes introduites au système d'indexation.

Lorsque l'utilisateur introduit l'image requête 1 dans notre prototype, le système calcule la distance de similarité entre chaque image individuelle de la vidéo de la base de données et l'image requête et extrait les images individuelles les plus similaires à cette image requête, ces images similaires peuvent appartenir aux différents plans, si parmi les distances calculées, n'existe pas une distance nulle alors le système affiche à l'utilisateur que l'image ne figure pas dans la vidéo.

L'utilisateur a la possibilité de voir les détails de la recherche, c'est-à-dire les images individuelles les plus similaires à la requête et les plans auxquels elles appartiennent.

Table 1. Détails de recherche pour l'image requête 1

N_PL	N_IMG	DISTANCE
0	1	0.375
2	26	0.375
4	51	0.375
4	52	0.375
6	76	0.375
6	77	0.375
9	101	0.375

Pour l'image requête 2, le système procède de la même manière, il trouve que la distance de similarité est nulle et dans ce cas là il affiche à l'utilisateur que l'image figure dans la vidéo.

Table 2. Détails de recherche pour l'image requête 2

N_PL	N_IMG	DISTANCE
3	33	0

En fait ces résultats sont obtenus avec la valeur de 0.9 du seuil, nous avons testé notre prototype sur le même exemple mais avec des valeurs différentes du seuil.

Table 3. Nombre d'images clefs en fonction des valeurs du seuil

Seuil	Nombre d'images clefs
0.1	65
0.2	60
0.3	53
0.4	43
0.5	31
0.6	25
0.7	20
0.8	15
0.9	11

Avec un petit seuil le nombre d'images clefs est relativement grand car un petit changement du contenu d'une image à une autre provoque la définition d'une nouvelle image clef, mais plus le seuil est grand moins il y a de chances de tomber sur une image qui varie beaucoup du contenu au point de définir un nouveau plan ce qui amène à avoir moins d'images clefs.

Un bon résumé de vidéo ne veut pas dire moins d'images clefs possibles, pour notre exemple un résumé de 2 images clefs ne donnerait pas un aperçu significatif et général de la vidéo ; en contre partie un résumé de 65 images clefs (avec un seuil de 0.1) exclut complètement la notion de résumé, ce n'est plus un résumé mais plutôt une partie de la vidéo, tout dépend en fait du contenu de la vidéo.

5 Conclusion

A travers cet article nous avons proposé une approche de recherche et d'indexation de vidéos par le contenu en utilisant la structure d'arbre R. Le prototype développé comporte deux principales phases : phase off-line qui consiste à indexer toutes les vidéos de la base par cette structure ce qui correspond à un autre codage de la vidéo et phase on-line dans laquelle l'utilisateur a la possibilité d'effectuer des recherches fines au sein de ces vidéos ; en effet dans cette phase l'utilisateur introduit une image ou une vidéo requête et le système l'indexe selon notre structure. Maintenant que les vidéos (y compris la vidéo requête) sont représentées par la même structure, le principe consiste à calculer la distance de similarité et d'extraire de la base les vidéos similaires à la vidéo requête selon le seuil qui a été défini.

Références

1. M. Scuturici "Contribution aux techniques orientées objets de gestion des séquences vidéo pour les serveurs Web". Thèse de Doctorat en Informatique. Institut National des Sciences Appliquées de Lyon. 2002.
2. M. Manouvrier "Objets similaires de grande taille dans les bases de données". Thèse de Doctorat en Informatique. Université Paris IX-Dauphine. 2000.
3. H. Abed : "Représentation et stockage des images similaires par des structures arborescentes". Mémoire de Magister en Informatique. Université des Sciences et de la Technologie d'ORAN Mohamed Boudiaf. 2002.
4. A. Guttman. R-trees : "A Dynamic Index Structure For Spatial Searching". Dans proc . Of ACM SIGMOD Int .Symp. On the management of data, pages 45-57, 1984.
5. http://www.infres.enst.fr/people/saglio/etudes/Strasbrg.html
6. M. Rukoz, M. Manouvrier, G. Jomier "Distance de similarité d'images basées sur les arbres quaternaires". Dans CNRS – CONICIT. 2002. pages 1 à 20.
7. http://www.lis.inpg.fr/pages_perso/chassery/francais/recherche.html
8. http://www.afrif.asso.fr/archive/rfia2004/ARTICLES.html
9. http://bat710.univ-lyon1.fr/ligim/textes/equipes/IMAGE/CORESA03/articles/html
10. F. Souvannavong : "Indexation et recherche de plans vidéo par le contenu sémantique". Thèse de Doctorat en Informatique. Ecole Nationale Supérieur des Télécommunications (Paris). 2005.

La reconnaissance en-ligne des caractères isolés par les réseaux bayésiens dynamiques: segmentation en strokes élémentaires

Redouane Tlemsani et Abdelkader Benyettou

SIgnal-IMage-PArole (SIMPA)
Dépt. Informatique, Université des Sciences et de la Technologie d'Oran – USTO, Algérie
tlemsani_redouane@yahoo.fr, aek.benyettou@univ-usto.dz

Résumé: Le travail se situe dans le domaine de la reconnaissance de caractères isolés en ligne. Ce système est un système Markovien que l'on peut voir comme un Réseau Bayésien Dynamique. Un des intérêts majeurs de ces systèmes réside dans l'apprentissage complet des modèles (topologie et paramètres) à partir de données d'apprentissage.
Les réseaux bayésiens sont des modèles représentant des connaissances incertaines sur des phénomènes complexes. Ils sont une union entre la théorie des probabilités et la théorie des graphes afin de donner des outils efficaces pour représenter une distribution de probabilités jointes sur un ensemble de variables aléatoires. La représentation de la connaissance se base sur la description, par des graphes, des relations de causalité existant entre les variables définissant le domaine d'étude.
La théorie des DBNs est une généralisation des réseaux Bayésiens aux processus dynamiques. Notre objectif revient à trouver une meilleure structure qui représente les liens (dépendances) entres les variables d'un réseau bayésien dynamique. Dans des applications en reconnaissance des formes, on va procéder à la fixation de la structure qui nous oblige d'admettre quelques hypothèses fortes (par exemple l'indépendance entre certaines variables).

Mots clés: reconnaissance de caractères isolés en ligne, reconnaissance des formes, réseau bayésien dynamique.

1 Introduction

Depuis les années soixante, l'Homme cherche à "apprendre à lire" aux ordinateurs. Une tâche de reconnaissance est difficile pour les caractères manuscrits isolés car leurs formes sont plus variées par rapport les caractères imprimés. La reconnaissance *en-ligne* permet d'interpréter une écriture représentée par la trajectoire du stylo.

Cette technique est notamment utilisée dans les agendas électroniques de type PDA. Une tablette électronique et un stylo spécial sont nécessaires. Le signal est capté en temps réel. Il est constitué d'une suite de coordonnées de points, correspondant à la position du stylo à intervalles réguliers de temps. En effet, le signal en-ligne contient des informations dynamiques absentes dans les signaux hors-ligne, telles que l'ordre dans lequel les caractères ont été formés, leur sens de parcours, la position des poser et lever de stylo.

Pour que la reconnaissance de caractères isolés soit fortement précise, il est important aussi de modéliser la structure des caractères ordinairement que possible. Dans ce mémoire nous considérons qu'un caractère se compose de strokes et on a gardé même leurs relations spatiales. Les strokes sont les éléments conceptuels et leurs relations spatiales sont conceptuellement significatives et qui sont habituellement robustes contre des variations géométriques et importantes pour les caractères distinctifs des formes semblables.

Un réseau bayésien peut modéliser des dépendances entre plusieurs variables aléatoires d'une façon probabiliste et graphique représentative. On utilise les réseaux bayésiens pour leur capacité de représenter une distribution gaussienne de probabilités conditionnelles jointes sur un ensemble de variables aléatoires.

2 Réseaux bayésiens

Les réseaux bayésiens (Bayesian Networks BNs) (également connu sous le nom de *réseaux de croyance*, *modèles génératifs*, *modèles causals*, etc.…) sont les modèles graphiques probabilistes qui codent des rapports probabilistes sur les graphes acycliques dirigés (Directed Acyclic Graphs DAG).Les rapports probabilistes sont définis comme ensemble d'affirmations conditionnelles de l'indépendance (Conditional Independence CI).La structure graphique code les relations conditionnelles de l'indépendance. La paramétrisation numérique indique la forme fonctionnelle de la représentation basée sur ces relations. Les méthodologies statistiques conçues dans ce formalisme fournissent des techniques informatiques efficaces pour les deux aspects de la représentation, c.-à-d. structure graphique et paramétrisation numérique.

Un BN est construit par ensemble de nœuds, où chaque nœud est associé à une variable aléatoire X_i. Les indépendances conditionnelles parmi les variables sont représentées avec un graphe acyclique dirigé (DAG).
Cette factorisation permet l'utilisation des distributions conditionnelles locales simples plutôt que d'employer le plein JPD « Joint Probabilisty Distribution » pour impliquer des probabilités.Un réseau bayésien est un pair (S,θ), où S est un graphe acyclique dirigé DAG, θ est une paramétrisation pour l'ensemble $\{P(X_1|\Pi_1),.....,P(X_n|\Pi_n)\}$ des distributions des probabilités conditionnelles, un pour chaque variable, et Π_i est l'ensemble des parent pour le noeud X_i dans S. L'ensemble pour CPDs « Conditional Distribution Probability» définit l'association JPD comme suit:

$$P(X)=\prod_{i=1}^{n} P(X_i|\Pi_i) \tag{1}$$

3 Réseaux bayésiens dynamiques

Les réseaux bayésiens dynamiques (DBN) prolongent la représentation de BN aux processus dynamiques. Un DBN code le JPD d'un ensemble d'évolution de temps $X[t]=\{X_1[t]...., X_n[t]\}$ de variables. En d'autres termes, il représente la croyance au sujet de la trajectoire possible du processus dynamique $X[t]$. Après une notation

semblable avec la représentation statique de BN, le JPD pour un intervalle fini de temps *[1,T]* est factorisé comme :

$$p(X[1],.....,X[T]) = \prod_{t=1}^{T}\prod_{i=1}^{n} P(X_i[t]|\Pi_i[t]) \quad (2)$$

Où $\prod_i[t]$ dénotent les parents de $X_i[t]$ dans le graphique structurent du DBN. La structure graphique d'un DBN peut être regardée comme concaténation des plusieurs BNs statique lié avec les *arcs temporels*. Nous appelons chacun de ces réseaux statiques une tranche de temps (Une tranche de temps est définie comme collection de l'ensemble de variables X[t] dans un seul temps t instantanée et leurs parents associés $\prod[t]$ dans la structure de graphique) du DBN. Dans le cas le plus général, si aucune prétention ne sont imposées au processus dynamique fondamental, la structure du graphe et la paramétrisation numérique d'un DBN peut être différente pour chaque fois découpent en tranches. Dans ce cas-ci le DBN est considéré comme BN (statiques) avec $T \times n$ variables et le codage du JPD peut être extrêmement complexe.

3.1 Représentation

Dans la littérature la représentation de DBN le plus souvent est employée pour les premiers processus stationnaires de Markov d'ordre. Pour ce cas, Friedman et autres décrit une représentation simplifiée en termes de deux charge statique BNs définie au-dessus des variables d'une tranche simple de temps [2]. Le principe de la représentation se fonde sur la prétention de stationnarité qui implique que la structure et les paramètres du DBN répètent. Le JPD est codé en utilisant un premier réseau et un réseau déroulé de transition.

Le réseau initial code la structure irrégulière dans la frontière et indique les états initiaux d'excédent de distribution X[1]. Le réseau de transition code la probabilité temps invariable de transition $P(X[t+1]|X[t])$. Le JPD pour un intervalle fini de temps est obtenu en déroulant le réseau de transition pour un nombre suffisant de tranches de temps. Le mécanisme de déroulement se compose présenter un ensemble de variables pour chaque fois découpent en tranches et replier la structure et les paramètres du réseau de transition sur ces variables. Réarrangeant les limites dans Eq.(*) le JPD est factorisé au-dessus des réseaux d'initiale et de transition comme :

$$p(X[1],.....,X[T]) = P_{B_1}(X[1]) \prod_{t=2}^{T-1} P_{B\rightarrow}(X_i[t]|X[t-1]) \quad (3)$$

Là où $P_{B_1}(.)$ et $P_{B\rightarrow}(.)$ sont les densités de probabilité codées par les réseaux d'initiale et de transition, respectivement.

3.2 Inférence dans DBNs

Le problème d'inférence dans DBNs est analogue au problème d'inférence de BN tels que désiré la quantité est la distribution marginale postérieure d'un ensemble de variables cachées indiquées un ordre des observations (mise à jour de croyance) : $P\left(X_h[t] \middle| X_0[1],...,X_0[\tau]\right)$ Où $X[t]=\left\{X_h[t],X_0[t]\right\}$ est un ensemble de variables d'évolution de temps dans lesquelles $X_o[t]$ et $X_h[t]$ dénotent des variables observées et cachées, respectivement. L'inférence de la série chronologique désigné généralement sous le nom filtrage $(\tau=1)$, lissant $(\tau>1)$ et du prévision $(\tau<1)$ selon la fenêtre de temps d'observation utilisée dans les calculs.

Une approche directe pour impliquer des probabilités dans un DBN, est construire un BN statique énorme pour le nombre désiré de tranches de temps et puis d'employer les algorithmes généraux d'inférence pour BNs statique. Cependant, ceci exige que la fin de l'ordre de temps soit connue a priori. En outre, la complexité informatique de cette approche peut extrêmement exiger (particulièrement en termes de mémoire). Par conséquent en général, l'inférence de DBN est exécutée en utilisant les opérateurs récursifs qui mettent à jour l'état de croyance du DBN pendant que les nouvelles observations deviennent disponibles. Le principe est semblable au message passant l'algorithme pour BNs statique. L'idée est aux messages définis sur une couverture de Markov des variables qui d-sépare le passé du futur et emploie un procédé vers la procédure forward- backward pour distribuer toute l'évidence le long du DBN [2,3, 4]Cette technique exige seulement d'une fenêtre de temps des variables d'être maintenue dans la mémoire. Ces algorithmes sont en effet généralisation de l'algorithme (Baum-Welch) vers *forward-backward* bien connu [5] dans les HMMs et cas spéciaux de l'algorithme de JLO [6].

4 Modélisation

Dans cette partie, nous considérons qu'un caractère se compose de strokes et de leurs relations spatiales. Les strokes sont des tracés élémentaires directs ou presque droits qui ont des directions distinctes des tracés reliées dans l'ordre d'écriture. Les relations spatiales des strokes indiquent les dépendances des positions entre les strokes ; la position d'un stroke obtient une influence sur les autres strokes.

4.1 Modèle de stroke (réseau bayésien statique)

Un exemple de stroke se compose de points. Par conséquent, un modèle de stroke se compose de modèles de point avec leurs relations spatiales, appelé relations spatiales intra strokes « Within Stroke Relationships » (WSRs).
Une WSR est définie comme dépendance d'un point médian entre deux points finals d'un stroke. Le point médian est le point d'où la longueur de stroke partielle gauche et ce de la droite sont égales.
La figure 1 montre l'exemple récursif de construction de stroke. À la première itération récursive (d = 1), IP_1 est ajouté pour modéliser des points médians de tous les exemples de stroke. Elle a la WSR des points finals (les arcs d'EP_0 et d'EP_1 à IP_1). À la deuxième itération récursive (d = 2), IP_2 et IP_3 sont ajoutés pour des points médians des strokes partielles gauches et droites, respectivement. En outre, elles ont

les WSR des points finals des strokes partielles. Figue.6 (c) est le modèle prolongé de stroke.
Avec ce procédé récursif, un modèle de stroke peut avoir autant de modèles de point selon les besoins. Dans ce chapitre, la profondeur de récursivité d=3 est choisie pour tous les modèles de stroke.
Elle vaut la peine de noter que les modèles de point à de grandes profondeurs de récursivité n'encourent pas le problème de modèle non adéquat. Car lorsque la profondeur est grande, les strokes partielles deviennent beaucoup plus courtes et plus linéaires. Par conséquent, les WSRs deviennent beaucoup plus fortes et les probabilités jointes des modèles supplémentaires de point obtiennent plus près la probabilité d'un seul. La probabilité jointe est obtenue à partir de ceux des modèles de point. Supposons qu'un modèle S a la profondeur d et un exemple de stroke a des points de longueur t : O(1),…,O (t). Pour assortir, l'exemple de stroke est périodiquement prélevé dans les 2^{d-1} points médians. Ils sont dénotés comme IP_1, $IP_{2....,}$ IP_{2d-1} selon l'ordre du procédé de prélèvement récursif.
Puis, les IPi d'exemples de point sont assortis aux modèles l'IPi de point. La probabilité jointe est calculée comme suit par la propriété locale de Markov des probabilités conditionnelles dans les réseaux bayésiens :

$$\begin{aligned} & P\left(S=O(1),...,O(t)\right) \\ & =P\left(\begin{array}{l} EP_0=O(1), EP_1=O(t), \\ IP_1=ip_1,...,IP_{2^d-1}=ip_{2^d-1} \end{array}\right) \\ & =P\left(EP_0=O(1)\right)P\left(EP_1=O(t)\right) \\ & \quad \times\prod_{i=1}^{2^d-1}P\left(IP_i=ip_i \setminus pa\left(IP_i\right)\right) \end{aligned} \tag{4}$$

Où le pa(IPi) est la configuration des noeuds parents lesquels les arcs de dépendance viennent à IPi.

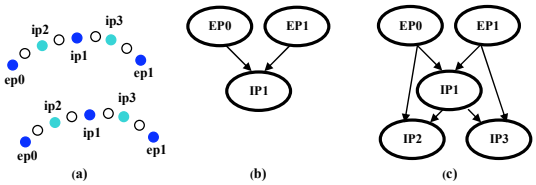


Fig. 1. *La construction récursive d'un modèle de stroke.*
(a) Exemple des ip1's : point médian de stroke ip2s et ip3's :
ceux des strokes partielles gauches et droits.
(b) modèle de stroke de la profondeur d = 1.
(c) Modèle de stroke pour d=2.

4.2 Modèle de caractère (réseau bayésien dynamique)

Un exemple de caractère se compose des strokes. En outre, les relations étroites existent entre elles. Par conséquent, un modèle de caractère se compose des modèles

de stroke avec leurs relations spatiales, appelées relations spatial inter strokes ou en anglais « Inter Stroke Relationships (ISRs) ». Les ISRs sont représentées avec des dépendances des points finals des strokes. Dans le meilleur des cas, un stroke obtient l'influence de tous les points d'autres strokes.

Dans La Figue.2, EP_0 est le premier modèle de point écrit dans un caractère. Les modèles de point de premier stroke sont écrits dans l'ordre d'$IP_{1,2}$, $IP_{1,1}$, $IP_{1,3}$. Puis, les modèles de point des deuxièmes strokes sont écrits dans l'ordre d'EP_1, $IP_{2,2}$, $IP_{2,1}$, $IP_{2,3}$. Alternativement, les strokes suivants sont écrits de la même façon. En conclusion, EP_N est le dernier modèle de point écrit dans un caractère [7,8,9].

Le modèle de vraisemblance d'un caractère est calculé par l'énumération de toutes les segmentations possibles de stroke. Supposons qu'un modèle BN de caractère a N modèles de stroke et une entrée de caractère avec T points : O(1),…,O(T). Puisque l'entrée n'a pas l'information de frontière, différentes segmentations sont possibles. On pose un exemple de segmentation de stroke par $\gamma = (t_0, t_1, \ldots, t_N)$, $t_0 = 1 < t < \ldots < t_N = T$, et l'ensemble de totalité par Γ. Puis le modèle de vraisemblance d'un caractère est donné comme suit :

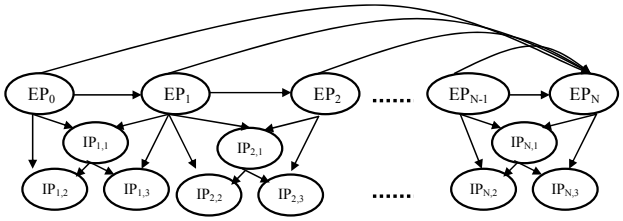


Fig. 2. *La représentation par réseau bayésien d'un modèle de caractère avec N strokes et la profondeur de récursivité d = 2.*

$$\begin{aligned} &P(O(1),\ldots,O(t)\setminus BN) \\ &= \sum_{\gamma=(t_0,\ldots,t_N)\in\Gamma} P(S_1 = O(t_0,t_1),\ldots,S_N = O(t_{N-1},t_N)) \\ &= \sum_{\gamma\in\Gamma}\prod_{i=1}^{N} P(S_i = O(t_{i-1},t_i)\setminus S_1 = O(t_0,t_1),\ldots,S_{i-1} = O(t_{i-2},t_{i-1})) \\ &= \sum_{\gamma\in\Gamma}\prod_{i=1}^{N} P(S_i = O(t_{i-1},t_i)\setminus EP_0 = O(t_0),\ldots,EP_{i-1} = O(t_{i-1})) \end{aligned} \tag{5}$$

Où $O(t_i, t_j) = O(t_i), O(t_{i+1}),\ldots,O(t_j)$. La probabilité jointe donnée par des strokes précédents est calculée comme suit :
Où $ip_{i,j}(O(t_{i-1}, t_i))$ est le $j^{ième}$ point échantillon de $O(t_{i-1}, t_i)$. En subtituant Eq. (5) pour Eq. (6), la probabilité du modèle ce n'est qu'un produit des probabilités jointes d'EPs et d'IPS :

$$P(S_i = O(t_{i-1},t_i)\setminus EP_0 = O(t_0),\ldots,EP_{i-1} = O(t_{i-1})) = \begin{cases} P(EP_i = O(t_i)\setminus O(t_0),\ldots,O(t_{i-1})) \\ \prod_{j=1}^{2^d-1} P(IP_{i,j} = ip_{i,j}(O(t_{i-1},t_i))\setminus pa(IP_{i,j})) \quad if\ i>1, \\ P(EP_0 = O(t_0))P(EP_1 = O(t_1)\setminus O(t_0)) \\ \prod_{j=1}^{2^d-1} P(IP_{i,j} = ip_{i,j}(O(t_{i-1},t_i))\setminus pa(IP_{i,j})) \quad if\ i=1, \end{cases} \tag{6}$$

$$P\left(O(1),...,O(t)\setminus BN\right)= \sum_{\gamma\in\Gamma}\prod_{i=0}^{N}P\left(EP_i=O(t_i)\setminus O(t_0),...,O(t_{i-1})\right) \times\prod_{i=1}^{N}\prod_{j=1}^{2^d-1}P\left(IP_{i,j}=ip_{i,j}\left(O(t_{i-1},t_i)\right)\setminus pa\left(IP_{i,j}\right)\right) \tag{7}$$

Les probabilités jointes des EPs peuvent être interprétées par des probabilités des positions de stroke globales et ceux des IPs avec des probabilités des formes de stroke locales.

5 Reconnaissance et apprentissage

5.1 Algorithme de reconnaissance

Un caractère manuscrit est identifié en trouvant le modèle de caractère qui produit la probabilité a posteriori la plus élevée donnée l'entrée. Quand le i$^{\text{ième}}$ modèle de caractère est dénoté BNi et les points d'entrée en tant que O(1)…, O(T), alors le problème de reconnaissance peut être formulé comme suit :

$$\begin{aligned}&\arg\max_i P\left(BN_i\setminus O(1),...,O(T)\right)\\ &=\arg\max_i\frac{P(BN_i)P\left(O(1),...,O(T)\setminus BN_i\right)}{P\left(O(1),...,O(T)\right)}\\ &=\arg\max_i P(BN_i)P\left(O(1),...,O(T)\setminus BN_i\right)\end{aligned} \tag{8}$$

La vraisemblance du modèle de caractère est décrite précédemment dans (équation (9)). Pour le calculer, toutes les segmentations possibles de stroke Γ sont considérées. Pour empêcher la complexité exponentielle de temps, nous supposons qu'elle peut être rapprochée par la probabilité jointe de caractère de la segmentation la plus probable γ^* dans Γ comme suit :

$$P\left(O(1),...,O(T)\setminus BN_i\right)\approx \max_{\gamma\in\Gamma}P\left(S_1=O(t_0,t_1),...,S_N=O(t_{N-1},t_N)\right) \tag{9}$$

Pour effectuer le calcul de la vraisemblance du modèle maniable dans le temps, nous avons besoin de plus une prétention pour la recherche de γ^*. En assortissant un stroke, toutes les segmentations possibles de ses strokes précédents devraient être considérées en raison des dépendances d'inter-stroke. Pour la simplicité de la recherche, nous supposons que la probabilité jointe d'un stroke est la plus haute avec la configuration la plus probable des strokes précédents. Ensuite, l'algorithme de programmation dynamique de recherche ou en anglais « *dynamic programming search algorithm* » peut être adopté comme suit :[10,11,12]

S_i **:** $i^{ième}$ modèle de stroke
$\gamma_i(t)$: la segmentation la plus probable quand S1,…..,Si et O(1,t) sont matchés.
$\delta_i(t)$: la probabilité jointe donnée $\gamma_i(t)$.
Initialisation
$\delta_0(1)=1$, $\gamma_0(1)= \{\}$
Stroke matching
for t=2 *to* T
for i=1 *to* N
$\delta_i(t) = max_{1\le b<t} P(S_i = O(b,t) \setminus \gamma_{i-1}(b)) . \delta_{i-1}(b)$
$b^* = argmax_{1\le b<t} P(S_i = O(b,t) \setminus \gamma_{i-1}(b)) . \delta_{i-1}(b)$
$\gamma_i(t) = \gamma_{i-1}(b^*) \cup \{t\}$
end
end
Vraisemblance du modèle de caractère
$P(O(1),....,O(T) \setminus BN_i) \approx \delta_N(T)$

5.2 Algorithme d'apprentissage

Pour construire un modèle de caractère, sa structure de dépendance et les paramètres conditionnels de probabilité devraient être déterminés. Dans ce chapitre, la structure de dépendance est déterminée par un concepteur modèle à partir de la connaissance a priori et les expériences. La profondeur de récursivité des modèles de stroke est choisie égale à trois(d=3). Le nombre de modèles est déterminé à partir du nombre typique de stroke dans le caractère.

Les paramètres conditionnels de probabilité sont formés par des données d'apprentissage. Ils sont les matrices de régression linéaire W's ($W=[w_{i,j}]$) et les covariances Σ's pour les modèles de points (équations 6) et (7)). Si tous les modèles de point sont assortis aux exemples de point, alors ils peuvent être estimés à partir des algorithmes statistiques conventionnels de régression avec l'objet de maximum de vraisemblance ML « maximum likelihood » [13]. Supposons qu'un point P dépend de $P_1 \ldots, P_k$ et il y a N échantillons d'apprentissage. On note le $i^{ième}$ échantillon de P as p(i) et les valeurs de variable dépendante par $z^{(i)} = [x^{(i)}_1 , y^{(i)}_1 ,\ldots, x^{(i)}_k , y^{(i)}_k , 1]$. Puis, ils sont estimés comme suit: [13]

$$W = \left(\sum_{i=1}^{N} p^{(i)} \left(z^{(i)} \right)^T \right) \left(\sum_{i=1}^{N} z^{(i)} \left(z^{(i)} \right)^T \right)^{-1} \tag{10}$$

$$\Sigma = \frac{1}{N} \sum_{i=1}^{N} p^{(i)} \left(p^{(i)} \right)^T - \frac{1}{N} W \sum_{i=1}^{N} z^{(i)} \left(p^{(i)} \right)^T \tag{11}$$

Pendant l'apprentissage du modèle de caractère, la re-estimation des paramètres et la recherché de la segmentation en strokes la plus probable γ^* est répété alternativement [11,12,13]

Cette approche est semblable à l'algorithme d'apprentissage EM (Expectation-Maximization). Etant donné les paramètres (W et Σ), γ* est mis à jour. Ensuite, avec la nouvelle γ*, les paramètres sont re-estimés. L'algorithme détaillé est comme suit:

- *Etape 1*: Initialiser le modèle de caractère avec les données initiales (une partie des exemples des strokes manuellement segmentés).
- *Etape 2* **:** Chercher la segmentation la plus probable γ* de la totalité des caractères d'apprentissage non segmentés en utilisant l'algorithme de recherché précédente.
- *Etape3* : Estimer les paramètres (W et Σ) sur les exemples partitionnés par les γ*.
- *Etape4* : Répéter les étapes 2 et 3 jusqu'à que la somme des vraisemblances du modèle ne changera plus (stabilité).

6 Expériences, résultats et analyses

6.1 corpus utilisé

La base UNIPEN est la base de données utilisée pour la réalisation de nos expériences. Il s'agit d'une base de grande taille, très employée dans l'évaluation des performances des différents systèmes de reconnaissance d'écriture.
La base UNIPEN est la base de référence pour l'élaboration et la comparaison de systèmes de reconnaissance d'écriture. Nous utilisons ici la partie de cette base contenant des caractères isolés, chiffres, lettres minuscules et lettres majuscules. Cette base contient des tracés de plus de 200 scripteurs. La difficulté de cette base est due principalement au nombre de scripteurs et donc aux nombreux allographes qu'ils emploient. Dans ces expériences, nous disposons de la version R01/V06.

6.2 Segmentation manuelle des caractères isolés

Notre étude concerne la reconnaissance en-ligne c'est-à-dire acquise dynamiquement a l'aide d'une tablette graphique a digitaliser. Celle-ci possède une résolution précise et échantillonne a une vitesse d'écriture choisie. un temps d'adaptation est nécessaire au scripteur pour pouvoir écrire a peu près correctement. malgré tout pour son traitement le caractère devra être segmente en trace c'est-à-dire en strokes élémentaires. La figure ci-dessus montre les différents changements implémentés et appliqués par exemple sur le caractère "A".

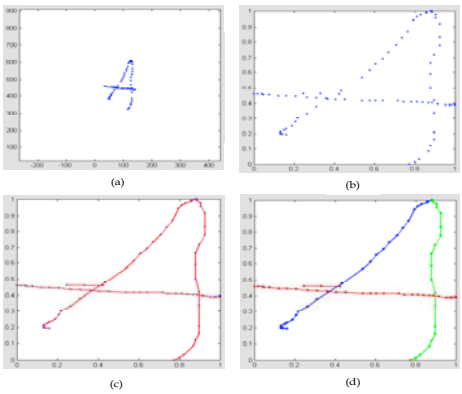


Fig. 3. Traitement et segmentation du caractère "A" (a) caractère original (b) caractère normalisé(c) caractère lié par la trajectoire du stylo (d) caractère segmenté.

6.3 Modèle bayésien dynamique

Pour nos expériences on a implémenté et vérifier notre modèle bayésien sur un ensemble de lettres du A à H de l'alphabet latin. La table 1 montre la matrice de confusion des taux de reconnaissance et ce sont des taux relativement importants sur un tel sous corpus et c'est une efficacité revient aux éléments spatiales sauvegardés et aux vraisemblances des gaussiennes de chaque modèle de point.Le taux global de reconnaissance atteint 70,24%.

Table 1. Modèle bayésien: Taux de reconnaissance des lettres du A à H.

%	*A*	*B*	*C*	*D*	*E*	*F*	*G*	*H*
A	**98,33**	**-**	**-**	**-**	**-**	**01,66**	**-**	**-**
B	**-**	**97,50**	**-**	**-**	**-**	**02,50**	**-**	**-**
C	**-**	**-**	**44**	**-**	**44**	**02**	**-**	**-**
D	**-**	**-**	**-**	**100**	**-**	**-**	**-**	**-**
E	**-**	**-**	**-**	**-**	**70**	**20**	**-**	**10**
F	**-**	**-**	**-**	**-**	**28,15**	**69,55**	**-**	**2,21**
G	**03,22**	**-**	**-**	**-**	**25,81**	**38,71**	**29,03**	**3,22**
H	**08,88**	**-**	**-**	**-**	**11,11**	**26,66**	**-**	**53,33**

Les résultats présentent des taux augmentés pour les lettres A, B et D. des taux relativement acceptables pour les lettres E et F. et des taux moins inférieur et égal à 50 pour les lettres C, G et H.

6.4 Vérifieur neuronal

La méthode implémentée dans ce mémoire est un modèle probabiliste qui se résume au concept des réseaux bayésiens dynamiques et après avoir obtenu γ^* le vecteur des vraisemblances par l'algorithme de programmation dynamique, au lieu de prendre la valeur maximale, les chercheurs en reconnaissance des formes utilisent souvent un vérifieur à ce niveau pour adapter le système. Les réseaux de neurones sont les classificateurs les plus utilisés dans le domaine de l'intelligence artificielle et on a tenté à intégrer ce mécanisme discriminant comme outil de vérification des formes.

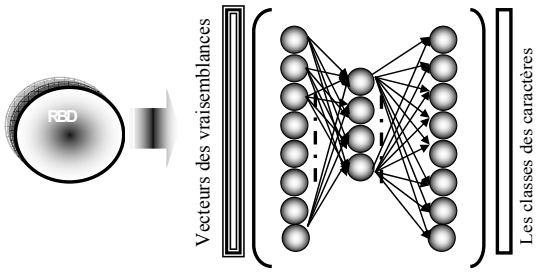


Fig. 5. *: Vérifieur neuronal*

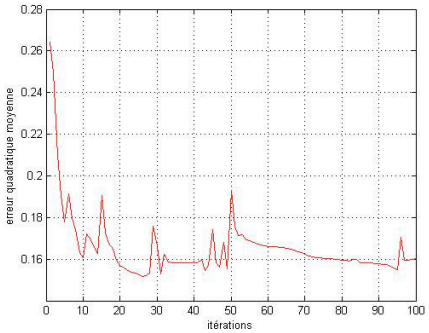


Fig. 6. *: Graphe de l'évolution de l'erreur par itération*

L'architecture neuronale utilisée est montrée dans la figure ci-dessus (8 neurones en entrée, 4 dans la couche cachée et 8 neurones dans la couche de sortie), en entrée affectation des vecteurs des vraisemblances obtenus par le réseau bayésien dynamique. En sortie un vecteur binaire indique la classe de caractère associé. Dans la figure 6, on remarque clairement la convergence de l'erreur quadratique moyenne obtenue à partir d'un apprentissage neuronale.

La table ci-dessous montre la matrice de confusion des taux de reconnaissance pour le modèle bayésien avec un vérifieur neuronal et on remarque au niveau de quelques caractères tels que le A et le B il y a une performance totale, mais il existe aussi des caractères non déterminés tels que 55% de la lettre C et 65% de la lettre E et cela revient au mécanisme probabiliste des réseaux bayésiens. Le taux global de reconnaissance atteint 66.87%.

%	*A*	*B*	*C*	*D*	*E*	*F*	*G*	*H*
A	**100**	**-**	**-**	**-**	**-**	**-**	**-**	**-**
B	**-**	**100**	**-**	**-**	**-**	**-**	**-**	**-**
C	**-**	**-**	**15**	**-**	**-**	**-**	**85**	**-**
D	**-**	**-**	**-**	**100**	**-**	**-**	**-**	**-**
E	**-**	**-**	**-**	**-**	**-**	**25**	**65**	**10**
F	**-**	**-**	**-**	**-**	**-**	**70**	**30**	**-**
G		**-**	**-**	**-**	**-**	**5**	**90**	**-**
H	**20**	**-**	**-**	**-**	**-**	**10**	**10**	**60**

Table 2. *Vérifieur neuronal: Taux de reconnaissance des lettres du A à H.*

6.5 Algorithme de recherche dynamique ARD par plage de points

L'idée de cet algorithme est de partager le signal d'écriture en des tranches de points, alors que cet algorithme donne une segmentation au niveau de chaque tranche de points, l'ancien algorithme consiste à affecter au niveau de chaque point du signal, une segmentation de ce dernier, ce qui est très lent pour un système de reconnaissance en ligne. Les deux approches:

- **Plage de points statiques :** On fixe un nombre de points statiques pour toutes les observations, (exemple : le nombre total des points du signal T=48, la plage P=3, l'ancien algorithme ARD se boucle 47 fois pour donner la segmentation finale alors que la nouvelle boucle exige que le nombre ((T/P)+1) est arrondi aux entiers proches supérieurs)

- **Plage de points dynamiques :** On fixe ici un pourcentage de points au lieu d'un nombre fixe pour toutes les observations, ainsi le nombre de points des plages se change selon le nombre T total des points d'une observation. (Exemple : le nombre total des points du signal T=48, le pourcentage des points Pr=5% alors la plage P=((T. Pr)/100), ce nombre est arrondi aux entiers proches supérieurs). L'idée de ce nouveau concept se résume dans la figure 7.

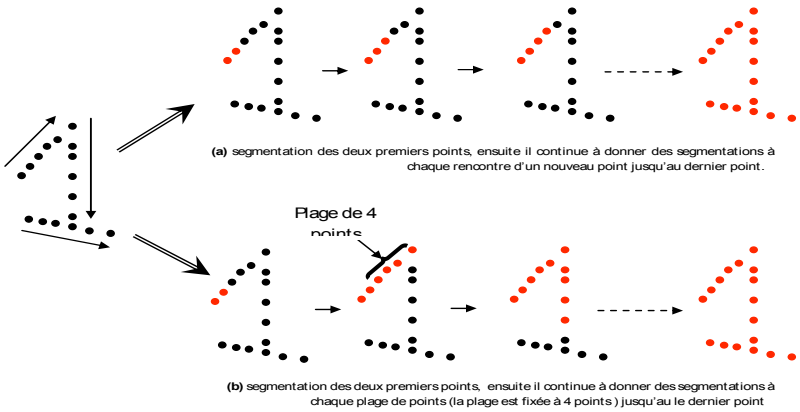


Fig. 7. *: Algorithme de recherche dynamique (ARD) par plage de points*

On a implémenté les trois approches (ARD, ARD par plage de points statiques et ARD par plage de points dynamiques) appliquées sur l'ensemble
des caractères de A à H et en sortie on a obtenue le temps de segmentation de l'observation indiquée (voir table 3). Le résultat montre efficacement que la méthode de segmentation basée sur la plage de points statiques est plus rapide par rapport à l'ancien algorithme ARD. Aussi cette approche par plage de points a montré une rapidité mieux remarquée globalement. On peut démontrer cette efficacité à cause du concept de base qui donne une possibilité de partition par un ensemble de points et non pas un point unique.

Table 3. *Temps de segmentation des observations pour les trois algorithmes*

	ARD	ARD plage Statique	ARD plage Dynamique
« A »	**93.015 s**	**4.985 s**	**1.36 s**
« B »	**188.468 s**	**9.266 s**	**1.734 s**
« C »	**37.797 s**	**2.329 s**	**1.407 s**
« D »	**22.468 s**	**1.625 s**	**1.172 s**
« E »	**77.718 s**	**4.235 s**	**1.735 s**
« F »	**32.078 s**	**2.063 s**	**1.641 s**
« G »	**71.485 s**	**3.766 s**	**1.781 s**
« H »	**45.203 s**	**2.375 s**	**1.375 s**
Moyenne	**71.029 s**	**3.830 s**	**1.526 s**

7 Conclusion

Pour faciliter davantage l'utilisation des ordinateurs, de nouvelles approches ont été proposés et crées. Elles n'utilisent plus le clavier ni la souris mais un stylet relié à une tablette électronique. Finalement, les réseaux bayésiens ne sont pas les seules approches usitées dans cet axe de recherche et on peut citer d'autres à base modélisation comme les chaînes de Markov cachées (HMM) ou biens les séparateurs à vastes marges (SVM) dans un but de diminuer les taux d'erreurs.

Références

1. M. Deviren. Dynamic Bayesian networks for speech recognition. In Proceedings of *AAAI 2002, SIGART/AAAI* Doctoral Consortium, Edmonton, Canada, 2002.
2. G. Zweig. Speech Recognition with Dynamic Bayesian Networks. PhD thesis, University of California, Berkeley, Spring 1998.
3. N. Friedman, K. Murphy, and S. Russell. Learning the structure of dynamic probabilistic networks. In *UAI'98, Madison*, Wisconsin, 1998.
4. K.P. Murphy. Dynamic Bayesian Networks : Representation, Inference and Learning. PhD thesis, UC Berkeley, Computer Science Division, 2002.
5. U. Kjaerulff. A computational scheme for reasoning in dynamic probabilistic networks. In Proceedings of *the Eighth Conference on Uncertainty in Artificial Intelligence*, pages 121-129, San Mateo, 1992. Morgan Kaufmann.
6. L.R. Rabiner and R.W. Schafer. *Digital Processing of Speech Signals*. Prentice Hall, Englewood Cliffs, NJ, USA, 1978.
7. F.V. Jensen, S.L. Lauritzen, and K.G. Olesen. *Bayesian updating in recursive graphical models by local computations*. Computational Statistics and Data Analysis, 1990.
8. K.Nathan, H. Beigi, J. Subrahmonia, G.J. Clary, H. Maruyama, Real-time on-line unconstrained handwriting recognition using statistical methods, *Proceedings of IEEE ICASSP*, Detroit, USA, Vol. 4, 1995, pp. 2619-2622.
9. T. Starner, J. Makhoul, R. Schwartz, G. Chou, On-line cursive handwriting recognition using speech recognition methods, Proceedings of *IEEE ICASSP,* Adelaide, Australia, Vol. 5, 1994, pp. 125-128.
10. B.-K. Sin, J. Kim, Ligature modeling for online cursive script recognition*, IEEE Trans. Pattern Anal*. Mach. Intell. 19 (6) (1997) 623-633.
11. H. Kim, J. Kim, Hierarchical random graph representation of handwritten characters and its application to Hangul recognition, *Pattern Recognition* 34 (2) (2001) 187-201.
12. C.-L. Liu, I. Kim, J. Kim, Model-based stroke extraction and matching for handwritten Chinese character recognition, *Pattern Recognition* 34 (12) (2001) 2339-2352.
13. I.-J. Kim, J. Kim, Statistical utilization of structural neighborhood information for oriental character recognition, Proceedings of *the Fourth IAPR International Workshop on Document Analysis Systems*, Rio de Janeiro, Brazil, 2000, pp. 303-312.

Series Expansions for Optimization of Maintenance in Stochastic Reliability Models

Fazia RAHMOUNE[1] and Djamil AISSANI[2]

[1] LAMOS Laboratory of Modelling and Optimization of Systems
University of Bejaia 06000, Algeria
foufourah@yahoo.fr,
[2] LAMOS Laboratory of Modelling and Optimization of Systems
University of Bejaia 06000, Algeria
lamos_bejaia@hotmail.com

Abstract. The work presented in this paper is a part of a major research project on numerical algorithms based on series expansions of Markov chains with finite state-space S. We are interested in the performance of a system when some of its parameters or characteristics are changed. The system as given is modeled as a Markov chain with kernel P, the changed system with kernel Q, having unique stationary distributions. The question is, can the deviation between stationary distributions be approximated or bounded in terms of the deviation (P-Q), for certain total variation norm. This is known as *Optimization Perturbation Analysis* of stochastic models in literature. The key idea of our approach is to solve certain Optimization Problem with Constraints. The present paper established the main theorical results, and the analysis provided applies to the case of optimization of preventive maintenance in reparable reliability models.

1 Introduction

Let $\mathbf{P}$ denote the transition kernel of a Markov chain defined on a finite state-space S having unique stationary distribution π_P. For example, think of $\mathbf{P}$ as the embedded jump chain of an $M/M/1/c$ queue, where c is a finite number denoting the buffer capacity of the queue. What would be the effect on the stationary behaviour of the queue if, for example, we increased the buffer capacity or adjusted the service rate of the queue? Let $\mathbf{Q}$ denote the Markov transition kernel of the Markov chain modeling the alternative system and assume that $\mathbf{Q}$ has unique stationary distribution π_Q. The question about the effect of switching from $\mathbf{P}$ to $\mathbf{Q}$ on the stationary behavior is expressed by $\pi_P - \pi_Q$, the difference between the stationary distributions. Let $\| \, . \, \|_{tv}$ denote the total variation norm, then the above problem can be phrased as follows: Can $\| \, \pi_P - \pi_Q \, \|_{tv}$ be approximated or bounded in terms of $\| \, \mathbf{P} - \mathbf{Q} \, \|_{tv}$? This is known as *perturbation analysis of Markov chains (PAMC)* in the literature.

In this paper we will show that $\pi_P - \pi_Q$ can be arbitrarily closely approximated by a polynomial in $(\mathbf{Q} - \mathbf{P})D_P$, where D_P denotes the deviation matrix

associated with $\mathbf{P}$, a precise definition will be given later on. Starting point is the representation

$$\pi_Q = \sum_{n=0}^{k} \pi_P((\mathbf{Q}-\mathbf{P})D_P)^n + \pi_Q((\mathbf{Q}-\mathbf{P})D_P)^{k+1}; \tag{1}$$

for any $k \geq 0$, a proof of which will be given in Section 3.1. This series expansion of π_Q provides the means of approximating π_Q by $\mathbf{Q}$ and entities given via the $\mathbf{P}$ Markov chain only. In order to obtain a bound for the remainder term, we propose to work with the weighted supremum norm, denoted by $\| \, . \, \|_v$, where v is some vector with positive non-zero elements, and for any $w \in \mathbb{R}^S$

$$\| \, w \, \|_v = \sup_{i \in S} \frac{w(i)}{v(i)}, \tag{2}$$

see, for example, [12]. We will show that

$$\pi_Q(s) - \left(\sum_{n=0}^{k} \pi_P((\mathbf{Q}-\mathbf{P})D_P)^n \right)(s) \leq d \, \| \, ((\mathbf{Q}-\mathbf{P})D_P) \, \|_v^{k+1};$$

for any $k \in \mathbb{N}$ and any $s \in \mathbb{S}$, where v can be any vector satisfying $v(s) \geq 1$ for $s \in \mathbb{S}$, and d is some finite computable constant. In particular, the above error bound can be computed without knowledge of π_Q.

The key idea of our approach is to solve for all k the optimization problem

$$\begin{cases} \min \| \, ((\mathbf{Q}-\mathbf{P})D_P)^k \, \|_v; \\ \text{subject to,} \qquad\qquad v(s) \geq 1 \quad \text{for} \quad s \in \mathbb{S}. \end{cases} \tag{3}$$

Surprisingly enough it will turn out that the solution $v^\star$ of the above optimization problem is independent of k. The vector $v^\star$ thus yields the optimal measure of the rate of convergence of the series in (1). Moreover, the series in (1) tends to converge extremely fast which is due to the fact that in many examples $v^\star$ be found such that $\| \, ((\mathbf{Q}-\mathbf{P})D_P)^k \, \|_{v^\star} << 1$. Best to our knowledge, the limit of the series (1) first appeared in [2], however, neither upper bounds for the remainder term nor numerical examples were given there. The use of series expansion for computational purposes is not new. It has been used in various fields for different purposes, for instance in the field of linear algebra [3]. In [1] the authors derive the power series for the stationary distribution in a slightly different way than [7] and the approach we take in this paper, but they don't use it for real problems or real computations. The novelty of our work is the fact that we combine the ideas obtained, and extend them such that problems can be really solved. The series expansion for the finite-state Markov chains in this paper is derived in a very elegant manner using the Poisson equation. We remark that this series expansion holds in a very general format under proper conditions. The derivation of this has been done in [7], which is a generalization

of [2].

The work presented in this paper is part of a major research project on numerical algorithms based on series expansions of Markov chains. The present paper establishes the main theoretical results. In a followup paper, we will study our methodology for large scaled problems. The paper is organized as follows. Section 2 presents basic facts on finite Markov chains. The series expansion (1) is discussed in Section 3. In particular, numerical examples are provided.

2 Preliminaries on Finite Markov chains

Let $\mathbb{S}$ denote a finite set with $0 < S < \infty$ elements. For notational convenience we will identify $\mathbb{S}$ with the set $\{1, \cdots, \mathbb{S}\}$. We consider Markov kernels on state space $\mathbb{S}$. Such a Markov kernel, say $\mathbf{P}$, can be written as square matrix $\mathbf{P} \in [0;1]^{S\times S}$. Element $(i;j)$ of $\mathbf{P}$ is denoted by $\mathbf{P}(i;j)$ and represents the probability of jump from state i to state j, which implies $\sum_{j\in\mathbb{S}} P(i;j) = 1$, for all $i \in \mathbb{S}$. The probability to go from state i to state j in n steps is denoted by $P_n(i;j)$, where the Markov kernel $\mathbf{P}^n$ is simply obtained from taking the nth power of $\mathbf{P}$. Provided it exists, we denote the unique stationary distribution of $\mathbf{P}$ by π_P and its ergodic projector by $\mathit{\Pi}_P$, that is, for any distribution μ it holds that $\mu\mathit{\Pi}_P = \pi_P$. In order to simplify the notation, we will with slight abuse of notation identify π_P and π_Q with $\mathit{\Pi}_P$ and $\mathit{\Pi}_P$, respectively. Throughout the paper we assume that $\mathbf{P}$ is aperiodic and unichain, which means that there is one closed irreducible set of states, and a (possibly empty) set of transient states. We write $|A|(i;j)$ to denote the $(i;j)$th element of the matrix of absolute values of $A \in \mathbb{R}^{S\times S}$, and additionally we use the notation $|A|$ for the matrix of absolute values of A.

The main tool for our analysis is the weighted supremum norm, also called v-norm, as defined in (2). For a matrix $A \in \mathbb{R}^{S\times S}$ the v-norm is given by

$$\| A \|_v =^{def} \sup_{i, \|w\|_v \le 1} \frac{\sum_{j=1}^{S} |A(i,j)w(j)|}{v(i)}$$

Observe that

$$\| A \|_v = \sup_i \frac{\sum_{j=1}^{S} |A|(i,j)v(j)}{v(i)}$$

which is due to $\| v \|_v = 1$ and the fact that $\| w \|_v \le 1$ implies $|w(i)| \le v(i)$, for $i \in \mathbb{S}$. Obviously, this implies

$$v(i) \| A \|_v \ge \sup_i \sum_{j=1}^{S} |A|(i,j)v(j), \quad i \in \mathbb{S}$$

and hence, using $v(i) \geq 1; i \in \mathbb{S}$, we obtain

$$\max_{j \in \mathbb{S}} |A|(i,j) \leq \| A \|_v \, v(i), \quad i \in \mathbb{S} \tag{4}$$

From (4) it readily follows that v-norm convergence to 0 implies elementwise convergence to 0. More precisely, let $\{A(n) \in \mathbb{R}^{S \times S}; n = 1; 2; \cdots\}$ be given such that $lim_{n \to \infty} \| A(n) \|_v = 0$ exists, then $lim_{n \to \infty} |A(n)|(i,j) = 0$ exists $\forall i, j \in \mathbb{S}$.

Next we introduce v-geometric ergodicity (also called v-normed ergodicity) of $\mathbf{P}_\theta$ at $\theta \in [0,1]$, see [12] for details.

Definition 1. *A Markov chain* $\mathbf{P}$ *is* v*-geometric ergodic if* $c < \infty, \beta < 1$ *and* $N < \infty$ *exist such that*

$$\| \mathbf{P}^n - \Pi_P \|_v \leq c\beta^n, \quad \textit{for all} \quad n \geq N.$$

The following lemma shows that any finite-state aperiodic Markov chain is v*-geometric ergodic.*

Lemma 1. *For finite-state and aperiodic* $\mathbf{P}$ *a finite number* N *exists such that*

$$\| \mathbf{P}^n - \Pi_P \|_v \leq c\beta^n, \quad \textit{for all} \quad n \geq N;$$

where $c < \infty$ *and* $\beta < 1$.

Proof. Because of the finite state space and aperiodicity,

$$\lim_{n \to \infty} \mathbf{P}^n(i,j) = \Pi_P(i,j); i, j \in \mathbb{S}.$$

Moreover, it is possible to take N sufficiently large such that

$$\forall i \in \mathbb{S} : \frac{\sum_{j=1}^{S} |\mathbf{P}^N - \Pi_P|(i,j) v(j)}{v(i)} < \epsilon;$$

with $\epsilon < 1$. Because $\mathbf{P}^n - \Pi_P = (\mathbf{P} - \Pi_P)^n$ for $n \geq 1$ this means that

$$\| (\mathbf{P} - \Pi_P)^N \|_v < \epsilon. \tag{5}$$

Taking appropriate integers n, k and l, such that $n = kN + l$ we then find

$$\| \mathbf{P}^n - \Pi_P \|_v \leq \epsilon^k \, \| \mathbf{P}^l - \Pi_P \|_v; \tag{6}$$

using norm inequalities. Define

$$c =^{def} \sup_{l=0,\cdots,N-1} \| \mathbf{P}^l - \Pi_P \|_v \tag{7}$$

and $\beta = \epsilon^{\frac{1}{N+1}}$. Since $\epsilon < 1$, it follows that $\beta < 1$, which implies

$$\epsilon^k = \beta^{kN+k} \leq \beta^{kN+l} = \beta^n, \tag{8}$$

where we use the fact that $0 \leq l < k$. Inserting 7 and 8 in 6, we obtain the stated.

We write D_P for the deviation matrix associated with $\mathbf{P}$; in symbols:

$$D_P = \sum_{m=0}^{\infty} (\mathbf{P}^m - \Pi_P) \tag{9}$$

Note that D_P is finite for any aperiodic finite-state Markov chain, see Lemma 2.2. Moreover, the deviation can be rewritten as

$$D_P = \sum_{m=0}^{\infty} (\mathbf{P} - \Pi_P)^m - \Pi_P,$$

where $\sum_{m=0}^{\infty} (\mathbf{P} - \Pi_P)^m$ is often referred to as the group inverse, see for in- stance [2, 4]. A general definition which is valid for any, possibly periodic Markov chain, can be found in, e.g., [14].

3 Series Expansions

We are interested in the performance of a system when some of its parameters or characteristics are changed. The system as given is modeled as a Markov chain with kernel $\mathbf{P}$, the changed system with kernel $\mathbf{Q}$. We assume that both Markov chains have a common finite state space S as defined earlier, i.e., $\mathbf{P}, \mathbf{Q} \in [0;1]^{S \times S}$. Note that sometimes phantom states have to be added to $\mathbf{Q}$ or $\mathbf{P}$ in order to achieve this; see Section 3.5 for an example. We also assume, as indicated earlier, that both Markov kernels are aperiodic and unichain. The goal of Section 3.1 is to obtain the stationary distribution of $\mathbf{Q}$, denoted by π_Q, via a series expansion in $\mathbf{P}$. In Section 3.2 we comment on the speed of convergence of this series. When applying these concepts to actually compute the stationary distribution, we have to solve the optimization problem stated in (3). We explain in Section 3.3 how to find an optimal solution. We summarize our results in an algorithm, presented in Section 3.4. Finally, we illustrate our approach with numerical examples in Section 3.5.

3.1 Series representation for π_Q

Let $\mathbf{P}$ be unichain (not necessarily finite). Elementary calculation shows (use the definition of D_P in (9)):

$$(I - \mathbf{P}) D_P = I - \Pi_P.$$

This is the Poisson equation in matrix format. Multiply this equation by Π_Q. Since $\mathbf{P}$ is unichain it holds $\Pi_Q \Pi_P = \Pi_P$, and we obtain $\Pi_Q (I - \mathbf{P}) D_P = \Pi_Q - \Pi_P$. Using that $\Pi_Q = \Pi_Q \mathbf{Q}$, we obtain

$$\Pi_Q = \Pi_P + \Pi_Q (\mathbf{Q} - \mathbf{P}) D_P. \tag{10}$$

Inserting (10) for Π_Q in the right-hand side of (10) we obtain

$$\Pi_Q = \Pi_P + \Pi_P(\mathbf{Q}-\mathbf{P})D_P + \Pi_Q((\mathbf{Q}-\mathbf{P})D_P)^2.$$

Repeating this step k times yields,

$$\Pi_Q = \Pi_P \sum_{n=0}^{k}((\mathbf{Q}-\mathbf{P})D_P)^n + \Pi_Q((\mathbf{Q}-\mathbf{P})D_P)^{k+1}. \tag{11}$$

for $k \geq 0$. Based on the above equation, we introduce the following notation. Let $k \geq 0$, then $H(k)$, with

$$H(k) =^{def} \Pi_P \sum_{n=0}^{k}((\mathbf{Q}-\mathbf{P})D_P)^n,$$

is called a *series approximation of degree k* for $\Pi_Q, T(k)$, with

$$T(k) =^{def} \Pi_P((\mathbf{Q}-\mathbf{P})D_P)^k, \tag{12}$$

denotes the kth element of H(k), and

$$R(k) =^{def} \Pi_Q((\mathbf{Q}-\mathbf{P})D_P)^{k+1}, \tag{13}$$

is called the *remainder term*. Notice that the remainder term $R(k)$ almost equals the (k + 1)st term of $H(k+1)$, i.e., $T(k+1)$, except for the prefactor, which is Π_Q in $R(k)$ and Π_P in $T(k+1)$. The quality of the approximation provided by $H(k)$ is given through the remainder term $R(k)$. This issue is discussed in the next section.

3.2 Convergence of the series

In this section we investigate the limiting behavior of $H(k)$ as k tends to ∞. We first establish sufficient conditions for the existence of the series.

Lemma 2. *The following assertions are equivalent:*

(i) The series $\sum_{k=0}^{\infty}((\mathbf{Q}-\mathbf{P})D_P)^k$ *is convergent.*
(ii) There are N *and* $\delta_N \in (0,1)$ *such that* $\| ((\mathbf{Q}-\mathbf{P})D_P)^N \|_v < \delta_N$.
(iii) There are κ *and* $\delta < 1$ *such that* $\| ((\mathbf{Q}-\mathbf{P})D_P)^k \|_v < \kappa\delta^k$ *for any* k.
(iv) There are N *and* $\delta \in (0;1)$ *such that*$\| ((\mathbf{Q}-\mathbf{P})D_P)^k \|_v < \delta^k$ *for any* $k \geq N$.

Proof. See [?].

Remark 1. Note that the fact that the maximal eigenvalue of $|(\mathbf{Q}-\mathbf{P})D_P|$ *is smaller than* 1 *is not necessary for* $\sum_{k=0}^{\infty}((\mathbf{Q}-\mathbf{P})D_P)^k$ *to converge, which stems from the fact that* $((\mathbf{Q}-\mathbf{P})D_P)^k$ *has positive and negative entries.*

Existence of the limit of H(k), see (i) in Lemma 3.1, is equivalent to an exponential decay in the v-norm of the elements of the series, see (iv) in Lemma 3.1. For practical purposes, one needs to identify the decay rate δ and the threshold value N after which the exponential decay occurs. We found in our numerical experiments that condition (ii) in Lemma 3.1 is the most convenient to work with. More specifically, we work with the condition (C).

(C) *There exists a finite number N such that we can find $\delta_N \in (0;1)$ which satisfies:*

$$\| ((\mathbf{Q}-\mathbf{P})D_P)^N \|_v < \delta^N;$$

and we set

$$c^v_{\delta_N} =^{def} \frac{1}{1-\delta_N} \| \sum_{k=0}^{N-1} ((\mathbf{Q}-\mathbf{P})D_P)^k \|_v$$

As shown in the following lemma, the factor $c^v_{\delta_N}$ in condition (C) allows to establish an upper bound for the remainder term that is independent of Π_Q.

Lemma 3. *Under (C) it holds that:*

(i) $\| R(k-1) \|_v \leq c^v_{\delta_N} \| T(k) \|_v$ *for all* k,

(ii) $\lim_{k\to\infty} H(k) = \Pi_P \sum_{n=0}^{\infty} ((\mathbf{Q}-\mathbf{P})D_P)^n = \Pi_Q$

Proof. To proof the lemma it is sufficient to use the definition of the norm $\| \, . \, \|_v$ and the remainder term $R(k-1)$, using the condition (iv) of Lemma 3.1.

Remark 2. An example where the series $H(k)$ fails to converge is illustrated in [?].

Remark 3. The series expansion for Π_Q put forward in Lemma 3.3 (ii) is well known; see [2] and [9] for the case of finite Markov chains and [7] for the general case. It is however worth noting that in the aforementioned papers, the series was obtained via a differentiation approach, whereas the representation is derived in this paper from the elementary equation (11).

Remark 4. Provided that $\det(I-(\mathbf{Q}-\mathbf{P})D_P) \neq 0$, *(10) determines π_Q uniquely and one can obtain π_Q from*

$$\pi_Q = \Pi_P (I-(\mathbf{Q}-\mathbf{P})D_P)^{-1} \tag{14}$$

Moreover, provided that the limit

$$\lim_{k\to\infty} H(k) = \lim_{k\to\infty} \pi_P \sum_{n=0}^{\infty} ((\mathbf{Q}-\mathbf{P})D_P)^n$$

exists (see Lemma 3.1 for sufficient conditions), it yields π_Q as $\pi_P \sum_{n=0}^{\infty} ((\mathbf{Q}-\mathbf{P})D_P)^n$; see Lemma 3.3. Equation (14) is the starting point for derivation of the series expansion put forward in Lemma 3.3 (ii) in [2].

Remark 5. Note that a sufficient (but not necessary) condition for (C) is

$$\| (\mathbf{Q} - \mathbf{P})D_P \|_v < \delta, \quad \delta < 1. \tag{15}$$

We will present examples for which it turns out that $\| (\mathbf{Q} - \mathbf{P})D_P \|_v > 1$, *although (C) is satisfied for* $N \geq 2$. *In [1, 3] it is even assumed that*

$$\| (\mathbf{Q} - \mathbf{P})D_P \|_v < g_1, \tag{16}$$

with $g_1 > 0$ *a finite constant, and*

$$\| D_P \|_v < \frac{c}{1-\beta}, \tag{17}$$

with $c > 0$ *and* $0 < \beta < 1$ *finite constants. If*

$$\frac{g_1 c}{1-\beta} < 1, \tag{18}$$

then (15) and hence (C) is clearly fulfilled. However, we can prove with some examples that we cannot find v *such that both (16), (17), and (18) hold. Hence, for numerical purposes these conditions are too strong.*

3.3 Bounding the remainder term

Until now we haven't specified v. *The quality of approximation by* $H(k-1)$ *is given by the remainder term* $R(k-1)$ *and in applications* v *should be chosen such that it minimizes* $c^v_{\delta_N} \| T(k) \|_v$, *thus minimizing our upper bound for the remainder term. Since* $c^v_{\delta_N}$ *is independent of* k, *we focus on* $T(k)$ *for finding an optimal upper bound. Specifically, we have to find a bounding vector* v *that minimizes* $\| T(k) \|_v$ *uniformly w.r.t.* k. *As the following theorem shows, the unit vector, denoted by* 1, *with all components equal to one, yields the minimal value for* $\| T(k) \|_v$ *for any* k.

Theorem 1. *The unit vector* 1 *minimizes* $\| T(k) \|_v$ *uniformly over* k, *i.e.,*

$$\forall k \geq 1 : inf_v \| T(k) \|_v = \| T(k) \|_1 \tag{19}$$

Remark 6. As for the results in [1, 3], it can be shown that the smallest g_1 *in (16) is the maximal eigenvalue of* $|(\mathbf{Q} - \mathbf{P})|$, *and the smallest* $\frac{cg_1}{1-\beta}$ *is precisely the maximal eigenvalue of* $|D_P|$. *Again we note that often the product of these maximal eigenvalues is not smaller than* 1. *If this is the case, then according to [1, 3] we cannot decide whether the series* $H(k)$ *converges to* Π_Q. *Hence, their condition is too restrictive for numerical purposes.*

3.4 Algorithm

In this section we describe a numerical approach to computing our upper bound for the remainder term $R(k)$. We search for N such that $1 > \delta_N =^{def} \| ((\mathbf{Q} - \mathbf{P})D_P)^N \|_1$, which implies that the condition (C) holds for N and δ_N. Then the upper bound for $R(k)$ is obtained from $c^1_{\delta_N} \| ((\mathbf{Q} - \mathbf{P})D_P)^{k+1} \|_1$. Based on the above, we can now describe an algorithm that yields an approximation for π_Q with ϵ precision. The algorithm has two main parts. First $c^1_{\delta_N}$ is computed. Then, the series can be computed in an iterative way until a predefined level of precision is reached.

Algorithm 1 *Chose precision $\epsilon > 0$. Set $k = 1, T(1) = \Pi_P(\mathbf{Q} - \mathbf{P})D_P$ and $H(0) = \Pi_P$.*

Step 1: Find N such that $\| ((\mathbf{Q} - \mathbf{P})D_P)^N \|_1 < 1$. Set $\delta_N = \| ((\mathbf{Q} - \mathbf{P})D_P)^N \|_1$ and compute

$$c^1_{\delta_N} = \frac{1}{1 - \delta_N} \| \sum_{k=0}^{N-1} ((\mathbf{Q} - \mathbf{P})D_P)^k \|_1 .$$

Step 2: If

$$c_\delta \| T(k) \|_1 < \epsilon,$$

the algorithm terminates and $H(k-1)$ yields the desired approximation. Otherwise, go to step 3.

Step 3: Set $H(k) = H(k-1) + T(k)$. Set $k := k+1$ and $T(k) = T(k-1)(\mathbf{Q} - \mathbf{P})D_P$. Go to step 2.

Lemma 4. *Provided that $\sum_{k=0}^{\infty} ((\mathbf{Q} - \mathbf{P})D_P)^k$ is finite, Algorithm 1 terminates in a finite number of steps.*

Proof. By Lemma 3.1, finiteness of $\sum_{k=0}^{\infty} ((\mathbf{Q} - \mathbf{P})D_P)^k$ implies (C). Thus, we know from Lemma 3.3 that $\| R(k) \|_1 \leq c^1_{\delta_N} \| T(k+1) \|_1$ for any k. Since

$$\| T(k+1) \|_1 \leq \| \Pi_P \|_1 \| ((\mathbf{Q} - \mathbf{P})D_P)^k \|_1,$$

the proof of the claim follows from the fact that $\lim_{k \to \infty} \| ((\mathbf{Q} - \mathbf{P})D_P)^k \|_1 = 0$, see Lemma 3.1.

3.5 Numerical Examples

The present paper established the main theorical results, and the analysis provided applies to the case of optimization of preventive maintenance in reparable reliability models.

Description of the models *Let us consider the $M/G/1//N$ reliability system with multiple preventives maintenances. We suppose that there is on the whole N machines in the system. Our system consists of a source and a waiting system (queue + service). Each machine is either in source or in the waiting system at any time. A machine in source arrives at the waiting system precisely, with the durations of inter-failure exponentially distributed with parameter λ) for repairment (corrective maintenance). The distribution of the service time is general with a distribution function $B(.)$ and mean b. The repairmen take maintenance period at each time the system is empty. If the server returns from maintenance finding the queue impty, he takes an other maintenance period (mulitiple maintenance). In addition, let us consider the $M/G/1//N$ reliability system with periodic prenventives maintenances, having the same distributions of the interarrivals and the repair time previously described. The server (repairman) will wait until the end of the next activity period during which at least a customer will be served, before beginning another maintenance period. In other words, there is exactly only one maintenance at the end of each activity period at each time when the queue becomes empty (exhaustive service). If the server returns from maintenane finding the queue nonempty, then the maintenance period finishes for beginning another activity period. We also suppose that the maintenance times V of the server are independent and iid, with general distribution function noted $V(x)$.*

Transition Kernels *Let X_n (resp. $\bar{X}_n$) the imbedded Markov chains at the end moments of repair t_n for the n^{th} machine associated with the $M/G/1//N$ system with multiple maintenance (resp.to the system with the unique maintenance). In the same way, we define the following probabilities:*

$f_k = P[k$ *broken down machines at the end of the preventive maintenance period*$]$

$$= C_N^k \int_0^\infty (1 - e^{-\lambda t})^k e^{-(N-k)\lambda t} dV(t), \ k = \overline{0, N}. \tag{20}$$

for witch

$$\begin{cases} \bar{\alpha}_k = \begin{cases} f_0 + f_1 & \text{for } k = 1, \\ f_k & \text{for } 2 \le k \le N. \end{cases} \\ \alpha_k = \dfrac{f_k}{1 - f_0} \qquad\qquad \text{for } k = \overline{1, N}. \end{cases}$$

The one stage transition probabilities of the imbedded Markov chains X_n and $\bar{X}_n$ allow us to describe the general expression of the transition kernels $\mathbf{P} = (\mathbf{P}_{ij})_{ij}$ and $\bar{\mathbf{P}} = (\bar{\mathbf{P}}_{ij})_{ij}$ summarized below respectively.

$$\mathbf{P}_{ij} = \begin{cases} \sum_{k=1}^{j+1} P_{j-k+1}\alpha_k, & \text{if } i = 0, j = \overline{0, N-1}, \ k = \overline{1, N}, \\ P_{j-i+1} & \text{if } 1 \le i \le j+1 \le N-1, \\ 0 & \text{else.} \end{cases} \tag{21}$$

and,

$$\bar{\mathbf{P}}_{ij} = \begin{cases} \sum_{k=1}^{j+1} P_{j-k+1}\bar{\alpha}_k, & \text{if } i = 0, j = \overline{0, N-1},\ k = \overline{1, N}, \\ P_{j-i+1} & \text{if } 1 \leq i \leq j+1 \leq N-1, \\ 0 & \text{else.} \end{cases} \tag{22}$$

Clearly, the Markov chain $\{\bar{X}_n\}_{n\in\mathbb{N}}$ *is irreducible, aperiodic with finite state space* $\mathbb{E} = \{0, 1, \cdots, N-1\}$. *So, we can applied the main theorical results established in this paper to this model, in order to approach another Markov chain whose transition kernel in neighborhood of its transition kernel* $\bar{\mathbf{P}}$. *Introducing a small disturbance on the structure of maintenance policy in* $M/G/1//N$ *system with multiples maintenances, we obtain the* $M/G/1//N$ *system with single maintenance policy (periodic maintenance). Then characteristics of this system can be approximated by those of the classical* $M/G/1//N$ *system with periodic maintenance, with a precision which depends on the disturbance, in other words on the maintenance parameter value.*

References

1. *E. Altman, K.E. Avrachenkov, and Nunez-Queija. Perturbation Analysis for Denumerable Markov Chains with Application to Queueing Models.* Advances in Applied Probability, Vol. 36:839-853, 2004.
2. *X.R. Cao. The Maclaurin Series for Performance Functions of Markov Chains.* Advances in Applied Probability, Vol. 30:676-692, 1998.
3. *G.E. Cho and C.D. Meyer. Comparison of Perturbation Bounds for the Stationary Distribution of a Markov Chain.* Linear Algebra and its Applications, Vol. 335:137-150, 2001.
4. *P. Coolen-Schrijner and E.A. van Doorn. The Deviation Matrix of a Continuous Time Markov Chain.* Probability in the Engineering and Informational Sciences, Vol. 16:351-366, 2002.
5. *G. Fayolle and R. Iasnogorodski. Two Coupled Processors: the Reduction to a Riemann-Hilbert Problem.* Zeitschrift fÄur Wahrscheinlichkeits theorie und verwandte Gebiete, Vol. 47:325-351, 1979.
6. *H. Hartmann and C. Arguelles. Transience Boundes for Long Walks.* Mathematics of Operations Research, Vol. 24:414-439, 1999.
7. *B. Heidergott and A. Hordijk. Taylor Series Expansions for Stationary Markov Chains.* Advances in Applied Probability, Vol. 35:1046-1070, 2003.
8. *A. Hordijk and N. Popov. Large Deviation Analysis of a Coupled Processor System.* Probability in the Engineering and Informational Sciences, Vol. 17:397-409, 2003.
9. *S. Kirkland. Conditioning Properties of the Stationary Distribution for a Markov Chain.* Electronic Journal of Linear Algebra, Vol. 10:1–15, 2003.
10. *C. Knessl and J.A. Morrison. Heavy Traffic Analysis of two Coupled Processors.* Queueing Systems: 173–220, 2003.
11. *A.G. Konheim, I. Meilijson, and A. Melkman. Processor Sharing of Two Parallel Lines.* Journal of Applied Probability, Vol. 18:952–956, 1981.
12. *S.P. Meyn and R.L. Tweedie. Markov Chains and Stochastic Stability. Springer, London, 1993.*

13. H. Minc. Nonnegative Matrices. Wiley, New York, 1988.
14. M.L. Puterman. Markov Decision Processes: Discrete Stochastic Dynamic Programming. John Wiley Sons, New York, 1994.
15. F. Rahmoune and D. Aissani. Approximation in Reparable Reliability Systems with Preventive Maintenance. Proceedings of the 5th. Multidisciplinary International Conference Quality and Dependability (RAMS) *(2005). Bordeaux–France, pp. 429–436.*
16. F. Rahmoune and D. Aissani. Approximation in Queueing Systems with Server Vacation. Proceedings of the " Colloque International sur l'Optimisation et les Systèmes d'Information (COSI'05), Béjaïa, Juin 12-14, 2005.
17. F. Rahmoune and D. Aissani. Quantitative Stability Estimates in Queues with Server Vacation. Journal of Mathematical Sciences, to appear. 2007

Gradual Uncertainty Association Rules Mining

Yassine Djouadi[1], Samir Redaoui[1], Karima Amroun[1].

[1]Department of Computer Science, University of Tizi-Ouzou
BP 17, RP, Tizi-Ouzou, Algeria
{ydjouadi,redaoui_samir,karima_amroun}@mail.ummto.dz

Abstract. Fuzzy sets have already been used for association rules mining problem when data is quantitative and precise. In such approaches fuzzy sets have been only used in order to partition a finite domain of quantitative and precise values into different intervals. These intervals are represented by means of fuzzy sets. We propose in this paper to enlarge this framework and consider imprecise and uncertain quantitative data represented by means of possibility distributions. For this purpose, a possibilistic association rules mining approach is presented in this paper. Considering possibility distributions upon fuzzy intervals will result on necessity/possibility measures. A general model using α-cuts is proposed in order to compute support and confidence regarding to necessity/possibility measures.

Keywords: Generalized Possibilistic Relational Model, fuzzy association rules, gradual uncertainty association rules.

1 Introduction

In many association rules mining problems data may be imprecise, vague and/or uncertain. Very few works in the community consider such data. Hence, the need for extending the general model of association rules mining process in order to take into account imprecision and/or uncertainty.

In an earlier paper [17], we have proposed to consider imprecise and/or uncertain data for association rules mining problems. Thus, we have proposed revisited measures of support and confidence. However our previous proposition doesn't take into account the semantics of induced rules.

For this purpose, we propose in this paper a more theoretical framework. We introduce first a new concept of "generalized possibilistic relational model". This model enlarges the well known "possibilistic relational model" [18] [5]. Consequently, the concept of "gradual uncertainty association rules" is also introduced in this paper. Such rules corresponds to the sound interpretation of fuzzy associations rules when data is imprecise.

The rest of this paper is organized as follows. In the next section, an overview of data mining, association rules mining is presented. The quantitative data mining problem is particularly detailed. In section 3, fuzzy association rules are presented and some corresponding works are presented. We summarize on this chapter and give the

main orientations of our contribution which is presented in section 4. The next section gives some results on synthetic data while the conclusion gives future perspectives.

2 Associations rules mining for quantitative data

There is still some confusion about the terms *Knowledge Discovery in Databases* and *Data Mining* [10] and often these two terms are used interchangeably. It seems then important to define more formally these two notions.

Knowledge Discovery in Databases (e.g. KDD) has been defined [9] as: "*the non trivial extraction of implicit, previously unknown, and potentially useful information from data*". While Data Mining consists of a set of tools and techniques used to implement a KDD process. Also, data mining is just one step in the overall KDD process.

Among these tools and techniques used in data mining, associations rules are one of the best model and one of the most studied. Association rules have aroused a growing interest in the past ten years regarding the numerous researches on this field [7], [11], [14], [20]. The framework of association was motivated by the desire to mine associations over sales transactions for market basket analysis. Also, the discovering process attempts to discover relationships among items such that the presence of certain items in a transaction (i.e. a tuple) tends to imply the presence of certain other items. Agrawal et al. [2] stressed the algorithmic aspects of the mining process at large. The first proposed approach was intended to binary attributes. Furthermore, the same authors [19] have proposed a method for mining association rules from datasets when data is quantitative or categorical. We present first the binary attributes case and then tackle the quantitative case and its inherent problems.

2.1 Binary attributes

We introduce now some of the formal terminology of association rules paradigm. A *transaction* is a set of items; in the original market basket scenario [2], the items of a transaction represent items that were purchased concurrently by a user. An *association rule* is a rule of the form $X \Rightarrow Y$ where X and Y are sets of items (or *itemsets*). An example of an association rule is "30% of transactions that contain Milk contain also Bread", 2% of all transactions contain both of these items. Here, the value 30% represents the confidence of the rule, and 2% represents the support of the rule.

More formally, let $\xi = \{i_1, i_2, ..., i_m\}$ be a set of items. Let $\psi = \{\tau_1, \tau_2, ..., \tau_n\}$ be a set of transactions (input dataset). An association rule is of the form $X \Rightarrow Y$ where $X, Y \subseteq \xi$, and $X \bigcap Y = \varnothing$ and $X, Y \neq \varnothing$. The support of $X \Rightarrow Y$ is defined as:

$$Supp(X \Rightarrow Y) = \frac{\left|\{\tau / \tau \in \psi \wedge X \bigcap Y \subseteq \tau\}\right|}{|\psi|}. \quad \textbf{(1)}$$

The confidence of the rule $X \Rightarrow Y$, denoted $Conf(X \Rightarrow Y)$, is defined as:

$$Conf(X \Rightarrow Y) = \frac{Supp(X \Rightarrow Y)}{Supp(X)}. \quad (2)$$

The problem of mining association rules is defined as: "given a set of transactions, a user-specified minimum support and a user-specified minimum confidence; find all associations rules with higher support and confidence than the specified threshold".

2.2 Quantitative and categorical attributes

The problem of mining association rules in large relational databases containing both quantitative and categorical attributes has been first presented in [19]. This problem is referred to as *quantitative association rules mining*. Authors propose to map a quantitative attribute A into a Boolean attribute in the following way:

Let t_i be a tuple (i.e. a transaction), let A be an attribute with quantitative values, let $t_i[A]$ denotes the value v of A in the tuple t_i (i.e. $t_i[A]=v$) and let $dom(A)$ represents the domain of values for attribute A. According to the relational paradigm, $dom(A)$ is necessarily a finite set. Two cases may are considered when mapping a Boolean attribute:

1. If the set $dom(A)$ consists of few values, the mapping is straightforward. Conceptually, authors propose to have instead of the attribute A, as many new attributes as the cardinality of the set $dom(A)$. The new attributes are of the form: $<A,v_k>$, where $v_k \in dom(A)$.

2. If the set $dom(A)$ is large, authors propose to partition this domain into different intervals $I_1,I_2,\ldots,I_p$. New attributes are of the form: $<A,I_k>$, where $dom(A) = I_1 \cup I_2 \cup \ldots \cup I_p$ and $I_i \cap I_j = \varnothing \quad \forall i,j \; i \neq j$.

However, two problems arise when partitioning a domain of values into many intervals. If the considered number of intervals for a quantitative attribute is important, the support for any single interval can be low. Hence, without using larger intervals, some rules involving this attribute may not be found because they lack minimum support. If the intervals are too large, some rules may not have minimum confidence. Indeed, when interval becomes larger, the support of the antecedent X of a rule $X \Rightarrow Y$ becomes higher. Consequently, the confidence of the rule (i.e. $Supp(X \cup Y)/Supp(X)$) decreases and the rule may not have minimum confidence.

Consequently if the intervals are too small some rules may not have minimum support; inversely if the intervals are too large, some rules may not have minimum confidence. This dilemma has led the community to consider intervals as fuzzy sets. This will be discussed in the next section.

3 Fuzzy intervals and fuzzy associations

Fuzzy sets theory [22] and possibility theory [21] supply a theoretical and formal framework to deal with imprecise and/or uncertain knowledge. Recently, the fuzzy set theory has been used more and more frequently in knowledge discovery systems [15]. To deal with the above mentioned dilemma, fuzzy approaches have been proposed for inducing association rules from crisp datasets especially when data is quantitative. We present first the general concepts of fuzzy sets theory and give an overview on fuzzy approaches for association rules mining.

3.1 Fuzzy sets: general concepts

Let X be a universe of discourse (a collection of objects) and x represents the generic element of X. A fuzzy set F in a universe of discourse X is a set characterized by a membership function μ_F which takes values in the interval [0,1]. Thus a fuzzy set F in X may be represented as a set of ordered pairs of a generic element x and its grade of membership function. When X is discrete, a fuzzy set F is represented as:

$$F=\sum_i \mu_F(x_i)/x_i$$

Let F and G be two fuzzy sets in a universe of discourse X, with membership function μ_F and μ_G respectively. The set theoretic operations of union and intersection for fuzzy sets are defined via their membership functions. More specifically, the membership function of the union $F \cup G$ (resp. the intersection $F \cap G$) is defined for all $x \in X$ by $\mu_{F\cup G}(x)=\mu_F(x) \oplus \mu_G(x)$ (resp. by $\mu_{F\cap G}(x)=\mu_F(x) \otimes \mu_G(x)$). Where the operator $\oplus$ (resp. $\otimes$) is a t-conorm (resp. t-norm). Well known example of t-conorm (resp. t-norm) is the maximum (resp. minimum) operator.

3.2 Fuzzy Association Rules Mining

Hong et al. [13] proposed a data mining algorithm for transaction data with quantitative values. This algorithm transforms the quantitative values in transactions into linguistic terms. Each item uses only linguistic terms with maximum cardinalities in later mining processes, thus making the number of fuzzy regions processed equal to the number of original items. The mined rules are expressed in linguistic terms which are more natural and understandable for human beings. The same authors [12] extended their approach to fuzzy association rules related to a taxonomy of items.

Delgado et al. propose to consider fuzzy cardinalities in order to determine fuzzy support [6]. A semantic approach is proposed. It is based on evaluation of quantified sentences of the form: Q of T are $\tilde{\tau}_A$, where Q is a relative fuzzy quantifier.

A fuzzy association rule mining system is also proposed by Au et al. [3]. Authors propose a system called FARM II (an earlier version called FARM is presented in [4]). This system considers quantitative and precise attributes. It employs also *linguistic terms* to represent the different intervals resulting from partitioning a domain. Interpretation of the linguistic terms is based on fuzzy sets theory. Authors emphasize on the use of appropriate linguistic representation.

It may be remarked that the above mentioned approaches consider only quantitative attributes which are precise and certain. We propose also in this paper to extend the quantitative association rules mining framework by considering imprecise and/or uncertain attribute values.

4 Gradual uncertainty association rules

Our previous proposition [17] enlarges the general framework of association rules mining by considering imprecise and/or uncertain data. However, we didn't take into account the semantics of imprecise data over fuzzy intervals (i.e. fuzzy attributes). For this purpose, we have to generalize first the concept of possibilistic databases by introducing the generalized possibilistic relational model.

4.1 The Generalized Possibilistic Relational Model

A possibilistic database is a particular case of imprecise database where imprecision is modeled using possibility theory [21]. More formally let $\{A_1, A_2, \ldots, A_m\}$ be a set of attributes (for the purpose of this paper, the domains D_j of attribute values will be considered as discrete set) and let $R(A_1, A_2, \ldots, A_m)$ denotes the classical relation schema. Hence, a *possibilistic relation* $\check{r}$ is defined on the schema R. An instance of $\check{r}$ is a subset of the Cartesian product of collections of possibility distributions [18]:

$$\check{r} \subseteq \Im(D_1) \times \Im(D_2) \times \ldots \times \Im(D_j) \times \ldots \times \Im(D_m)$$

where $\Im(D_j)$ denotes the collection of all possibility distributions of attribute A_j defined on domain D_j and $\times$ denotes the Cartesian product.

An example of possibilistic database is illustrated trough the relation given in table 1. In order to induce quantitative association rules in this example, the domain of attribute "*salary*" is partitioned in four fuzzy intervals, respectively S_1, S_2, S_3, S_4, (see fig. 1). These fuzzy intervals are represented using trapezoidal shapes. One may remark that no works exist in the literature in order to get an optimal fuzzy partition. In all related works [3], [8], [6], [12], [13], [15], [17] the shapes of fuzzy sets and the number of fuzzy intervals are empirically stated. In our approach, we intend to use optimal fuzzy partitions which are obtained using a fuzzy clustering method [1].

However, this method is not yet fully implemented also, in this paper; fuzzy partitions are given by an expert.

Table 1. Example of a possibilistic relation.

#tuple	Profession	Salary	Purchased Device
#1	engineer	50[1]	PC computer
#2	student	0.2/10+0.6/11+1/12 +0.6/13+0.2/	PC computer
#3	engineer	37	PC computer
#4	student	very low	CD-player
#5	farmer	0.2/48+0.6/49+1/50 +0.6/51+0.2/52	DVD-player
…	…	…	…
…	…	…	…

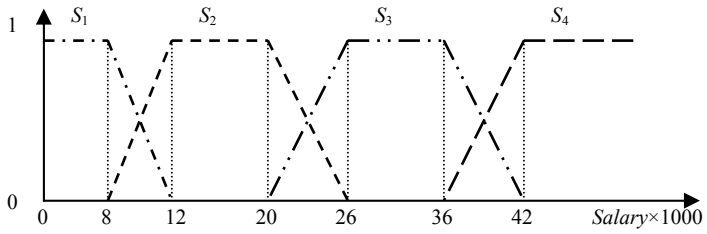


Fig. 1. Fuzzy partitions of the domain of salary attribute.

In order to consider both fuzzy intervals and imprecise attribute values, we propose to generalize the above defined possibilistic relational model. Let $\phi(D_j)$ represents the fuzzy partition of the domain D_j, h_j denotes the cardinality of the set $\phi(D_j)$ and $\tilde{f}_{jk}$ represents the *k*-th fuzzy set (fuzzy interval) of the attribute A_j

Proposition 1. A Generalized Possibilistic Relation $\aleph$ is defined on the schema $\mathcal{R}(\tilde{f}_{1,1},\tilde{f}_{1,2},\ldots,\tilde{f}_{1,h_1},\ldots,\tilde{f}_{j,k},\ldots,\tilde{f}_{m,1},\ldots,\tilde{f}_{m,h_m})$ as a subset of the Cartesian product of fuzzy restrictions of possibility distributions and is given as:

$$\aleph \subseteq (\Im(D_1)\,\wr\,\tilde{f}_{1,1})\times \ldots \times(\Im(D_j)\,\wr\,\tilde{f}_{j,k})\times\ldots\times(\Im(D_m)\,\wr\,\tilde{f}_{m,h_m}). \quad (3)$$

where $\wr$ denotes the fuzzy restriction operator.

[1] To abbreviate, values of salary are given of the form 50 instead of 50,000

Thus, semantics of the restriction operation ($\pi_{A(X)}(d) \wr \mu_{\widetilde{f}_{jk}}(d)$) represents the certainty value of a realization $A(X)$ over a fuzzy set $\widetilde{f}_{j,k}$. Given a generalized possibilistic relation $\aleph$ defined on the schema $\mathcal{R}(\widetilde{f}_{1,1},\widetilde{f}_{1,2},\ldots,\widetilde{f}_{1,h_1},\ldots,\widetilde{f}_{j,k},\ldots,\widetilde{f}_{m,1},\ldots,\widetilde{f}_{m,h_m})$, a *certain* interpretation of $\aleph$, denoted $\Omega(\aleph)$ correspond to the instance database containing only certainty values.

Let $\{t_1, t_2, \ldots, t_n\}$ be a set of tuples, v_{ij} denotes the precise value of attribute A_j in the tuple t_i (i=1..n, j=1..m), π_{ij} denotes the possibility distribution representing the imprecise value of attribute A_j in the tuple t_i and let $\mu_{\widetilde{f}_{jk}}$ denotes the membership function of the fuzzy set $\widetilde{f}_{jk}$ ($\mu_{\widetilde{f}_{jk}} : D_j \rightarrow [0,1]$).

Proposition 2. A tuple of $\aleph$ (denoted $\hat{t}_i$, i=1..n) corresponding to the certain interpretation $\Omega(\aleph)$ indicates certainty values (denoted $\hat{t}_i[\widetilde{f}_{jk}]$). If the attribute value is precise (i.e. $t_i[A_j] = v_{ij}$), certainty values are given as:

$$\hat{t}_i[\widetilde{f}_{jk}] = \mu_{\widetilde{f}_{jk}}(v_{ij}). \quad (4)$$

If the attribute value is imprecise (i.e. $t_i[A_j] = \pi_{ij}$) certainty values are given as:

$$\hat{t}_i[\widetilde{f}_{jk}] = \inf_{d \in D_j} \max(\mu_{\widetilde{f}_{jk}}(d), 1 - \pi_{ij}(d)). \quad (5)$$

4.2 Example of determining certainty values:

We consider the attribute *salary* given in Table 1 (the corresponding fuzzy partition is illustrated in figure 1). Certainty values are determined depending on the nature of the value (precise or imprecise). One may remark that our approach take into account different representations of imprecision. For instance, in the tuples #2 and #5, imprecision is modeled through a discrete distribution of possibility whereas a fuzzy quantified label (i.e. "*very low*") is considered in tuple #4. Depending also on the form of imprecision, different cases are considered.

Imprecise attribute values represented by a discrete possibility distribution

Let us illustrate the method for determining the certainty value for the tuple #2 (see also figure 2 and table 2). It may be seen that according to the fuzzy sets S_1 and S_2 respectively, two certainty values are obtained $\hat{t}_2[S_1]$ and $\hat{t}_2[S_2]$ respectively.

Table 2. Method for determining certainty values.

x	10	11	12	13	14
$\mu_{S_1}(x)$	0.5	0.25	0	0	0
$\pi_2(x)$	0.2	0.6	1	0.6	0.2
$1-\pi_2(x)$	0.8	0.4	0	0.4	0.8
$\max(\mu_{S_1}(x), 1-\pi_2(x))$	0.8	0.4	1	0.4	0.8
$\hat{t}_2[S_1]$		$\inf\{0.8,0.4,1,0.4,0.8\}=0.4$			
$\mu_{S_2}(x)$	0.5	0.75	1	1	1
$\max(\mu_{S_2}(x), 1-\pi_2(x))$	0.8	0.75	1	1	1
$\hat{t}_2[S_2]$		$\inf\{0.8,0.75,1,1,1\}=0.75$			

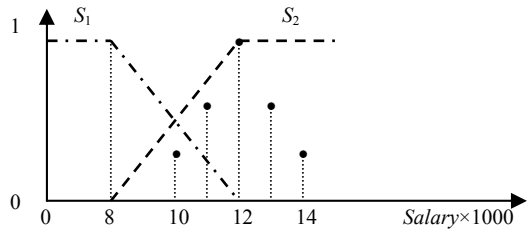


Fig. 2. Discrete distribution of possibility of salary attribute value for the tuple #2.

The values $\hat{t}_2[S_1]$ and $\hat{t}_2[S_2]$ are determined using expression (5). For instance, when x=10 then $\mu_{S_1}(10)$=0.5, $\pi_2(10)$=0.2 and, $1-\pi_2(10)$=0.8 and consequently: $\max(\mu_{S_1}(10), 1-\pi_2(10)) = \max(0.2,0.8)$. The table 2 illustrates the method.

Precise attribute values

A precise value v may be generalized by a distribution possibility which take the possibility value 1 when x=v and the value 0 otherwise. In such case, determining the certainty value is straightforward according to the expression (4).

Fuzzy labels and fuzzy quantified labels

Fuzzy labels and fuzzy quantified labels are a convenient way to represent imprecise attribute values. Fuzzy labels are completely defined by their fuzzy membership function. The possibility theory [21] is closely related to fuzzy set theory [22] since the idea is to constrain the values taken by a variable using a (normalized) fuzzy set called a possibility distribution. Also, the fuzzy membership function of the label is

considered itself as a (continuous) distribution of possibility. The figure 3 illustrates the distribution of possibility for the label "very low" corresponding to the value of *salary* in the tuple #4. The values $\hat{t}_4[S_1]$ and $\hat{t}_4[S_2]$ are also determined using expression (5). For instance, the dashed line in figure 3 indicates 1-$\pi_4(x)$ and we obtain:

$\hat{t}_4[S_1]=1$

$\hat{t}_4[S_2]=0$

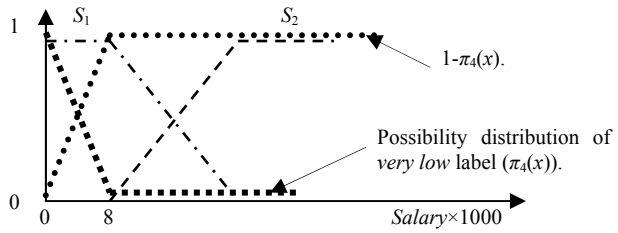


Fig. 3. Distribution of possibility of salary attribute value for the tuple #4.

4.3 Revisited measures of support and confidence

One semantics of discovered association rules is "the more *certainly X* is *A*, the more *certainly Y* is *B*". We propose to denote these rules as *gradual certainty association rules*. Consequently, associated measures of antecedent and consequent part for gradual certainty association rules are expressed in terms of necessity measures. The problem becomes the need to aggregate these measures.

Let P denotes the set of all fuzzy partitions (i.e. $P=\{\widetilde{f}_{j,k} / j=1..m \wedge k=1..h_j\}$) and let P_0 be a subset of P. To abbreviate we denote $P_0=\{f_1,f_2,...,f_k,...,f_p\}$.

Definition 1. Given a set P_0 then the α-level (denoted $\aleph_\alpha^{P_0}$) of a certain interpretation instance belonging to $\Omega(\aleph)$ is defined upon the set P_0 as:

$$\aleph_\alpha^{P_0}=\{\hat{t}_i, i=1..n / \min_{k=1,p}(\hat{t}_i[\widetilde{f}_k])\geq\alpha\}. \quad (6)$$

Related to a (crisp) α-level instance, we consider now the α-support of a fuzzy set P_0 (denoted $\lambda_\alpha(P_0)$) as a crisp support measure.

Proposition 3. The α-support of a fuzzy set P_0 ($P_0 \subseteq P$) is given by:

$$\lambda_\alpha(P_0) = \frac{\left|\aleph_\alpha^{P_0}\right|}{|\aleph|}. \quad (7)$$

Some irrelevances have been pointed out when using inappropriate support measure. In order to take more into account the contribution of itemsets with high certainty values, we have proposed in an earlier paper [17], a weighted summation of the cardinalities of the different α-support measures. Hence, given a set $\{\alpha_0, .., \alpha_i, .., \alpha_p\}$ (where $\alpha_0=0$, $\alpha_i<\alpha_{i+1}$, $\alpha_p=1$) the support of a fuzzy set is obtained through the α-supports of its α-cuts as follows:

$$Supp(P_0) = \sum_{i=1,p} (\alpha_i - \alpha_{i-1}).\lambda_{\alpha_i}(P_0) \ . \quad (8)$$

And the confidence measure of a rule $P_1 \Rightarrow P_2$ is easily obtained from (6) as:

$$Conf(P_1 \Rightarrow P_2) = \frac{\sum_i \alpha_i.\lambda_{\alpha_i}(P_1 \cap P_2)}{\sum_i \alpha_i.\lambda_{\alpha_i}(P_1)} \ . \quad (9)$$

Most association rules mining algorithm are fundamentally based around the anti-monotonic principle. This principle states that a super-set of an infrequent set is itself infrequent. This principle is used for pruning infrequent sets during the computational process. It may be proved that the α-support conserves this principle.

Proposition 4: Given a set P_0 ($P_0 \subseteq P$), a level α and, a threshold λ, if P_0 is infrequent at the level α (i.e. $\lambda_\alpha(P_0) < \lambda$), then all superset of P_0 is itself infrequent.

5 Experimental results

The proposed Possibilistic Association Rules Mining (P.A.R.M.) algorithm integrates the generalized possibilistic relational model and data mining concepts to find interesting rules from imprecise quantitative attribute values. The proposed PARM algorithm relies upon classical A-PRIORI implementation. The reader may refer to [17] for a complete implementation of PARM.

For our experiments, we have used syntactical data for instance, the database illustrated in table 1. We assume that the minimum support and confidence threshold are set respectively to 0.1 and 0.4. The table 4 gives some induced rules.

Table 4. Example of induced association rules.

Rules	Support	Confidence
Profession=*enginer* ⇒ Purchased device=*PC computer*	0.23	0.50
Profession=*enginer* ⇒ *P*urchased device=*CD Player*	0.16	0.53
Profession=*student* and Salary ∈ S_2 ⇒ *P*urchased device=*DVD Player*	0.13	0.56
….	…	…

Actual implementation of PARM integrates the anti-monotonic principle of the α-support measure. Indeed, we don't have to compute the support of a set P_0 at the level α_{i+1} if the set P_0 is itself infrequent at the level α_i (with $\alpha_i<\alpha_{i+1}$).

6 Conclusion

The main contribution of this work is to consider imprecise attribute values for quantitative association rules mining problem. The generalized possibilistic relational model is a powerful model able to handle different representations of incomplete data (possibility distributions, linguistic terms, fuzzy quantified expressions, …). Utilization of this model for quantitative association rules mining induces a new type of rule: gradual uncertainty association rules which generalizes fuzzy association rule type. This model supplies also a theoretical framework (i.e. fuzzy set and possibility theories) in order to achieve support and confidence measures.

Perspectives to this research are manifold. We intend to explore more deeply (through real world implementation) the support and confidence measures. Indeed, aggregating crisp measures may induce a lost of information. Also, we have to propose more suited measures.

Acknowledgments. This research was supported by the Ministry of High Education under research project *B*00520060029*.

References

1 Allouane, B., Djouadi, Y.: Recherche des sous ensembles flous par la méthode de classification non supervisée. Submitted to JNAM'07.
2 Agrawal, R., Imielinski, T., Swami, A.: Mining associations rules between sets of items in large databases. ACM SIGMOD Conference on Management of Data, Washington D.C. (1993) 207-216
3 Au, W.H., Chan, K.C.C.: Mining fuzzy association rules in a bank-account database. IEEE Transactions on Fuzzy Systems, Vol. 11, Issue. 2. (2003) 138-248
4 Au, W.H., Chan, K.C.C.: FARM: A data mining system for discovering fuzzy association rules. 8th IEEE International Conference on Fuzzy systems, Seoul Korea, (1999) 1217-1222

5 Bosc, P., Pivert, O.: About projection-selection-join queries addressed to possibilistic relational databases. IEEE Transactions on Fuzzy Systems, Vol. 13, Issue. 1. (2005) 124-139
6 Delgado, M., Marin, N., Sànchez, D., Vila, M.A.: Fuzzy association rules : General model and application. IEEE Transactions on Fuzzy Systems, Vol. 11, Issue. 2. (2003) 214-225
7 Djouadi, Y., Souam, F.: Bottom Up Text Mining trough Hierarchical Document Representation. International Conference on Knowledge Mining, ICKM'06, Prague, eds. Transactions on Engineering Computing and Technology (2006) 49-54.
8 Dubois, D., Prade, H., Sudkamp, T.: On the representation, measurement, and discovery of fuzzy associations. IEEE Transactions on Fuzzy Systems, Vol. 13, Issue. 2. (2005) 250-262
9 Frawley, W.J., Piatetsky-Shapiro G., and Matheus, C.J."Knowledge discovery in databases: an overview." *In G. Piatetsky-Shapiro and W.J. Frawley (edts), Knowledge Discovery in Databases, MIT Press, 1991*, pp. 1-27.
10 Goebels, M., Gruenwald, L.: A survey of data mining and knowledge discovery software tools. ACM SIGKDD Explorations, Vol. 1, Issue. 1. (1999) 20-33
11 Gouda, K., Zaki, M.: Efficiently mining maximal frequent itemsets. 1st IEEE International Conference on Data Mining, San Jose (2001)
12 Hong, T.P., Lin, K.Y., Wang, S.L.: Fuzzy data mining for interesting generalized association rules. Fuzzy Sets and Systems Journal, Vol. 138, Issue. 2. (2003) 255-269
13 Hong, T.P., Kuo, C.S., Chi, S.C.: A data mining algorithm for transaction data with quantitative values. Intelligent Data Analysis, Vol. 3, Issue. 5. (1999) 33-47
14 Park, J.S., Chen, M.S. Yu, P.S.: An effective hash-based algorithm for mining association rules. ACM SIGMOD Conference on Management of Data, San Jose (1995) 175-186
15 Pedrycz, W.: Fuzzy Set Technology in Knowledge Discovery. Fuzzy Sets and Systems Journal, Vol. 98, Issue. 3. (1998) 279-290
16 Prade, H., Testemale, C.: Generalizing database relational algebra for the treatment of incomplete/uncertain information and vague queries. Information Science, Vol. 34. (1984) 115-145
17 Redaoui, S., Djouadi, Y.: Association rules mining: application to imprecise data. LFA'06, Logique Floue et Applications Conference, Cepadues editor, Toulouse France (2006) 195-202
18 Rudensteiner, E.A., Bic, L.: Aggregates in possibilistic databases. Fifteenth Conference on Very Large Databases, Amsterdam Holland (1989) 287-296
19 Srikant, R., Agrawal, R.: Mining quantitative association rules in large relational tables. ACM SIGMOD Conference on Management of Data, Montreal Canada (1996) 1-12
20 Wang, J., Han, J., Pei, J.: CLOSET+: Searching for the best strategies for mining frequent closed itemsets. ACM SIKDD, Conference on Knowledge Discovery and Data mining, Washington D.C. (2003)
21 Zadeh, L.A.: Fuzzy sets as a basis for a theory of possibility. Fuzzy Sets and Systems, Vol. 1. (1978) 3-28
22 Zadeh, L.A.: Fuzzy Sets. Information and Control Journal. Vol. 8. (1965) 338–353

Application de la méthode des k-moyennes hiérarchique sur un corpus de textes arabes

Abdessalem Kelaiaia[1], Hayet Farida Merouani[2], Yamina Tlili-Guiassa[2]

[1] Département d'informatique, Université 08 Mai 1945,
Guelma, Algérie
(Kelaiaia Abdessalem)
[2]Département d'Informatique, Laboratoire LRI/Equipe SRF Sidi Ammar BP 12,
Annaba, Algérie
(Hayet Farida Merouani, Hayet_merouani@Yahoo.fr)
(guiam@Yahoo.fr)

Résumé
Dans cet article, nous proposons l'étude d'une approche de classification non supervisée pour une collection de documents en langue arabe. Nous exposons les objectifs visés par l'application du clustring dans les bases documentaires. La méthode des k-moyennes (k-means) initialisée par une hiérarchie est la méthode choisie pour notre expérimentation. Par l'application de cette méthode qui a été déjà testée sur un corpus en français et un autre en anglais, nous voulons étudier les résultats de son application pour un corpus en langue arabe, vue les particularités morphosyntaxiques de cette langue.

Mots-clés: Clustering, k-moyennes, classification hiérarchique, textes, langue arabe

1 Introduction

Il y a juste quelques années où les moteurs de recherche renvoyaient quelques pages Web pour une requête introduite. Dès lors, ce nombre a explosé depuis, des millions de pages apparaissent chaque jour, on ce retrouve devant un océan d'informations électroniques sous plusieurs formes tirées de plusieurs sources telles que les agences de presse, les news groups, les courriers électroniques, etc… Les utilisateurs tels les organismes commerciaux ou services, publics ou privés, les particuliers construisent à partir de cette source des bases documentaires gigantesques. Toute cette information serait sans intérêt si la capacité à y accéder efficacement n'est pas aussi fiable.

En effet, devant cette masse d'informations, la question d'accéder à l'information ne se pose plus, au contraire elle devient : comment trouver l'information dont on a besoin, parmi toutes celles qui sont accessibles ?

Le développement des outils qui permettent la recherche d'une information particulière, l'exploration de toute la collection, l'analyse de l'information et plus, tout ça en un temps raisonnable est inévitable. De tels outils rentrent tous dans le cadre de la fouille de textes (Text mining) et sont appelés systèmes de recherche documentaire (Documents Retrieval Systems) ils sont basés essentiellement sur des algorithmes de l'intelligence artificielle.

Dans cette optique, plusieurs conférences sont organisées périodiquement, la plus connue est la conférence internationale de la recherche de textes (Text REtrieval Conference, TREC) offrant un forum d'évaluation et de discussions pour la communauté scientifique qui se consacre au traitement automatique des textes dans différentes langues et récemment la langue arabe. Un des axes de recherche dans cette conférence et bien d'autres est la classification automatique. En effet la recherche d'une information dans une base documentaire ou son exploration sont deux tâches très délicates, si cette dernière n'est pas convenablement organisée. En faite, La performance d'un système de recherche documentaire est étroitement liée à une bonne organisation ou classification thématique de la source documentaire. La conception d'un outil qui, de façon automatique, assignerait ces textes à un ensemble prédéfini de classes est inévitable. C'est précisément le but de la classification automatique de textes.

L'objectif de cet article est l'étude d'une **approche des k-moyennes hiérarchique pour une collection de textes arabes,** en s'inspirons des travaux de Patrice Bellot et Marc El-bèze [Bellot et El-bèze, 2000], dans lesquels ils ont prouvé d'une part l'amélioration apportée par une classification automatique dans une recherche documentaire et d'autre part, apporté une réponse nouvelle au problème du choix délicat du nombre de classes (clusters) lors de l'utilisation d'une méthode des k-moyennes. Nous commencerons par donner une présentation à la classification automatique textuelle, nous donnerons ensuite quelques particularités de la langue arabe, suivie par l'approche choisie pour la représentation des textes, la cinquième section sera consacrée à la méthode que nous utiliserons. Nous conclurons sur notre méthodologie de recherche.

2 Classification automatique de textes

La classification automatique textuelle est le regroupement de textes dans des classes en utilisant des techniques statistiques qui produisent des résultats à partir des calculs de fréquence d'occurrence de termes extraits [Chauché et al, 2003]. Les techniques de la classification sont répertoriées selon deux principales approches, la première est dite classification supervisée, elle est basée sur l'apprentissage supervisé, la deuxième est dite classification non supervisée ou Clustering ou encore apprentissage non supervisée pour la classification[1], c'est dans cette dernière que notre étude se place.

Le Clustering a cumulé un regain d'intérêt surtout avec l'augmentation de l'intérêt apporté à l'information textuelle et l'abondance des textes électroniques qu'une base documentaire peut contenir. Employé conjointement avec un système de recherche documentaire, le Clustering aide, d'une part, les utilisateurs à évaluer rapidement la pertinence des documents retournés en réponse à une requête, ceci se base sur l'idée que si un document est pertinent à une requête, ses voisins (càd les documents similaires dans le même cluster) ont plus de chance d'être aussi pertinents [Fegas, 2005], et d'autre part, facilite la navigation à travers les clusters retournés.

Deux approches de classification non supervisée sont généralement utilisés [Jardino, 2004]: la classification hiérarchique (ascendante ou descendante) qui permet

[1] Dans la littérature, on ne retrouve pas une distinction signifiante entre les trois appellations.

de générer un arbre complet allant du regroupement de tous les éléments à classer dans une seule classe, à la racine de l'arbre, à la répartition de chaque élément dans sa propre classe, aux feuilles de l'arbre. L'autre méthode est le partitionnement direct en un nombre de classes spécifié à l'avance.

Quelque soit l'approche choisie, un processus de classification est précédé par une **phase de prétraitement** visant à normaliser la représentation des textes à classer. Divers outils tels que la radicalisation (stemming), la lemmatisation (limatization), l'indexation (indexing)… sont utilisés dans le traitement des langues anglaise, française, allemande et autres langues européennes. Pour plusieurs de ces langues, des techniques de classification et des règles élaborées ont été testées pour la recherche de l'information et l'extraction de connaissances. Pour la langue arabe quelques approches seulement fonctionnent. L'aspect **morphologique** très délicat de cette dernière qui n'apparaît pas dans les langues européennes, rend ces techniques difficilement applicables [Sawaf et al, 2001].

3 Particularité de La langue arabe

La langue arabe possède un alphabet qui compte 28 consonnes qui changent de forme de présentation selon leur position. Contrairement aux langues latines, l'arabe est une langue agglutinante; les articles, les prépositions et les pronoms collent aux adjectifs, noms, verbes et particules auxquels ils se rapportent ; ce qui engendre une ambiguïté morphologique au cours de l'analyse des mots [Boulaknadel, 2005]. Un mot arabe peut représenter une phrase en langue latine, il peut être composé d'un corps schématique (stem), des antéfixes qui sont des prépositions ou des conjonctions, des préfixes et suffixes qui expriment les traits grammaticaux et indiquent les fonctions : cas du nom, mode du verbe et les modalités (nombre, genre, personne,…) et des postfixes qui sont des pronoms personnels [Douzidia, 2004],[Diab et al, 2004]. Par exemple le mot **أتأكلونها**, qui veut dire : « Est-ce que vous là mangez ?», se décomposent comme suit :

Table 1, Décomposition d'un mot en arabe arabe

Postfixe	**Suffixe**	**Corps schématique**	**Préfixe**	**Antéfixe**
هَا	ونَ	أَكَلُ	تَ	أ
Pronom suffixe complément du nom	Suffixe verbal exprimant le pluriel	verbe	Préfixe verbal du temps de l'inaccompli	conjonction d'interrogation

Le collage des éléments flexionnels (antéfixes, préfixes, suffixes, postfixes) engendre des schèmes. La flexion d'un mot radical peut avoir jusqu'à 150 schèmes différents حمل→ محمول، محمل،حامل…. [Douzidia, 2004], ce qui rend l'application des techniques de prétraitement telle que la radicalisation, très utile surtout dans les systèmes de recherche d'information.

En langue arabe, les voyelles sont de deux types. Les voyelles courtes qui ont la forme d'une marque diacritique placée au dessus ou au dessous des lettres, et les voyelles longues collent aux consonnes. Les voyelles courtes sont une autre source d'ambiguïté quand elles sont supprimées (ce qui est le cas dans majorité des textes),

4 Indexation et représentation des documents

L'indexation consiste à associer à chaque document du corpus un ensemble de termes, chaque terme est appelé terme d'indexation (ou attribut). Le résultat de chaque document est un vecteur de termes d'indexation, avec lequel on pourra passer à la représentation vectorielle des documents. Dans notre étude, nous utiliserons pour représenter les documents, le formalisme vectoriel où chaque dimension de l'espace correspond à un terme d'indexation [Salton et al, 1975][Jaillet, 2003], ce dernier est pondéré par un poids. En résultat, on aura une matrice dont les lignes sont les documents de la collection et les colonnes sont les termes qui appartiennent à un vocabulaire V et qui comprend tout les termes apparaissant au moins une fois, dans la collection. Plusieurs pondérations ont été données aux poids d'un terme [Dunoyer, 2004], la plus connue est TF-IDF (Term Frequency, Inverse Document Frequency) qui est basée sur la loi de Zipf (attribuée à G.K. Zipf) qui stipule que **«un mot est informatif dans un document si il y est présent souvent mais qu'il n'est pas présent trop souvent dans les autres documents du corpus».** Cette représentation consiste à multiplier un facteur du poids du terme ti dans le document dj par un autre qui représente le poids du terme ti dans tout le corpus. Le modèle le plus classique pour calculer ces deux poids est que pour le premier on prend la fréquence du terme dans le document, notée tf_i^d et pour le deuxième on prend $\log\left(\frac{|D|}{df_i}\right)$ où |D| est le nombre de documents du corpus et df_i est le nombre de documents qui contiennent le terme i, en résultat on aura :

$$dji_{tf\text{-}idf} = tf_i^{dj} * \log\left(\frac{|D|}{df_i}\right) \quad \textbf{(1)}$$

5 Clustering par la méthode des k-moyennes initialisée par une hiérarchie

5.1 Méthode des k-moyennes

La méthode des k-moyennes (k-means) est sans doute la méthode de partitionnement la plus célèbre. Le principe est de choisir au hasard K points de l'espace qui seront un point de repère (centroïde) aux futurs clusters, on définie une fonction objective globale, qui est généralement la somme des distances des objets et les centroïdes, et itérativement, on déplace les objets à partitionner entre clusters dans le but d'optimiser cette fonction [Weiss, 2006]. Les clusters, prises deux à deux, doivent

être d'intersection vide, les centroïdes peuvent être un ou plusieurs objets (méthode des nuées dynamiques) ou un centre de gravité des objets du clusters [Turenne, 2000]. Le fonctionnement de la méthode des k-moyennes est décrit par l'algorithme suivant:

1- Choisir k représentant ou centroïde
2- Affecter chaque objet au centroïde le plus similaire
3- Recalculer le centroïde pour chaque cluster
4- Répéter les étapes 2 et 3 jusqu'à que le changement soit minime d'une passe à l'autre dans l'appartenance des membres aux clusters

Algorithme 1 : Méthode des k-moyennes

Le principal inconvénient de cette méthode est le choix du nombre k de clusters de la partition initiale.

5.2 Classification hiérarchique ascendante

Les algorithmes de la classification hiérarchique ascendante sont les plus connus des méthodes de la classification automatique [Turenne, 2000]. Ils aboutissent à un empilement de partitions dont le sommet réuni une partition et la base autant de partitions que d'objets à classer ; cet empilement peut se représenter par un dendrogramme (fig, 1). L'obtention d'une partition se fait par coupure du dendrogramme à un seuil donné.

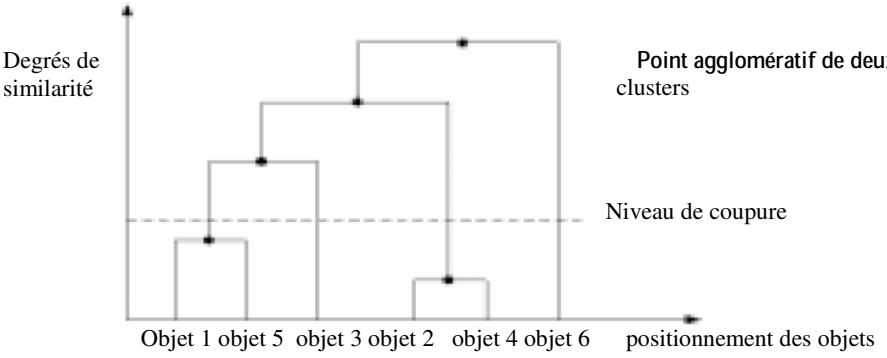


Fig.1. Forme type d'un Dendrogramme

Dans la recherche documentaire et malgré leur complexité algorithmique, les algorithmes de classification hiérarchique ascendante sont les plus utilisés. La construction d'une hiérarchie impose le calcul des distances séparant tous les documents pris deux à deux. Si le nombre de documents est trop élevé, il est impossible de conserver en mémoire la matrice des distances, Plusieurs solutions ont été alors proposées ; [Milligan & Cooper 1985] [Bellot et El-bèze, 2000] suggèrent d'imposer des critères d'arrêts durant la construction des hiérarchies tels que le nombre de clusters minimal. [Schütze & Silverstein, 1997] [Bellot et El-bèze, 2000] ne construisent une hiérarchie qu'à partir d'un sous-ensemble de documents choisis aléatoirement. Ils attribuent en fin de traitement chaque document non sélectionné cluster le plus proche [Bellot et El-bèze, 2000].

Le fonctionnement de la méthode d'une classification hiérarchique ascendante est décrit dans les étapes suivantes :

1. Associer chaque élément à classer à un nouveau cluster ;
2. calculer la matrice des distances séparant touts les clusters 2 à 2 ;
3. tant qu'il existe plus d'un cluster
   - Identifier les deux clusters les plus proches et les réunir dans un seul nouveau cluster,
   - Mettre à jour la matrice des distances.

5.3 Méthode des nuées dynamiques initialisée par une hiérarchie

La méthode de classification qui a été proposée dans [Bellot et El-bèze, 2000] repose sur les nuées dynamiques initialisée par une hiérarchie partielle. Les étapes suivantes décrivent l'algorithme utilisé :

1. Trouver une partition initiale
2. Faire :
   Calculer les représentants de chaque cluster
   Affecter chaque document au cluster le plus proche
   Jusqu'à stabilisation des clusters

Algorithme 2 : Méthode des nuées dynamiques initialisée par une hiérarchie

Dans ce qui suit nous détaillons les étapes de l'algorithme 2

Partition initiale

Etant donnée que la partition finale dépend fortement du choix de la partition initiale, cette dernière doit être choisie avec précaution, afin d'éviter le mauvais choix des représentants des clusters d'une part et d'autre part assurer la convergence de la méthode le plus vite possible. Pour cela, [Bellot et El-bèze, 2000] ont proposé de construire une hiérarchie partielle en deux étapes :

1. regroupement des documents séparés par une distance inférieure à un seuil préfixé : on choisit un seuil tel que la moitié des documents soit affectée à l'issue de cette étape. Si au cours de cette étape, avec di et dj deux documents, on a D(di,dj) < seuil, et que di (resp. dj) est déjà affecté à un cluster, on transfère di (resp. dj) et les autres documents de son cluster dans le cluster de dj (resp. di).
2. regroupement, deux à deux, des clusters les plus proches entre eux jusqu'à en avoir le nombre désiré. La distance entre deux clusters n'est pas la plus petite distance entre deux documents de chacun des clusters mais entre deux représentants de chacun des clusters. Ce procédé impose de conserver une matrice des distances séparant deux à deux touts les clusters.

Calcul des représentants d'un cluster

Le calcul des représentants d'un cluster (qui sont les documents les plus près du centre géométrique du cluster) est réalisé en n'utilisant que les valeurs des distances entre les documents: pour chaque cluster, les k documents centraux ou représentatifs

(k = 3 a été choisi empiriquement dans la compagne d'évaluation durant TREC-7 et Amaryllis'99) sont ceux dont les sommes des distances qui les séparent de tous les autres documents de la classe sont les plus petites.

Calcul de distance entre les documents

Soient D1 et D2 deux documents, u un terme, N(u) le nombre de documents contenant u dans la collection entière. Soit S le nombre total de documents dans la collection.

La quantité d'information associée à un terme u dépend du nombre de documents contenant u dans la collection entière (cette quantité est généralement dénommée IDF) :

$$I(u) = -\log_2 \frac{N(u)}{S} = IDF(u) \quad \textbf{(2)}$$

Soit $f_d(u)$, le nombre d'occurrences du terme u dans le document D.

La quantité d'information associée à un document est la somme des quantités d'information des termes du document :

$$I(D) = \sum_{u \in d} f_d(u) I(u) \quad \textbf{(3)}$$

La distance entre deux documents D1 et D2 est définie comme suit (avec D1 D2 désignant les termes communs aux deux documents) :

$$D(D1, D2) = 1 - \frac{I(D1 \cap D2)}{Max(I(D1), I(D2))} \quad \textbf{(4)}$$

Le calcul de la distance entre les documents doit assurer la convergence de la classification c'est-à-dire, l'utilisation d'une distance vérifiant notamment l'inégalité triangulaire, ce qui est le cas des trois équations (2), (3) et (4).

6 Méthodologie de recherche

Dans notre étude, nous avons utilisé le corpus compilé par Latifa El Sulaiti [Sulaiti, 2003] sur Radio Qatar collectionné pour le projet de l'arabe contemporain en Juin 2003. Il comporte 432 textes sous XML répartis en 16 clusters. Dans la première phase, qui est le prétraitement, nous avons procédé à :

-transformation de l'ensemble des textes du corpus en des textes plats (càd élimination des primitives XML) ;

-élimination des mots vides de sens (stop words) qui a permis une diminution considérable du vocabulaire ;

-application d'un traitement de radicalisation (stemming).

En résultat nous avons obtenu un vocabulaire de 32382 termes avec lesquels nous avons représenté les 432 textes sous la forme TF-IDF.

Dans la deuxième phase nous suivrons les étapes suivantes :

1. Application de la méthode des nuées dynamiques initialisée par une hiérarchie sur la représentation obtenue dans la phase de prétraitement,
2. Application de la méthode des centres mobiles initialisée par une hiérarchie sur la représentation obtenue dans la phase de prétraitement ;

3. Une comparaison, d'une part, entre les résultat obtenus dans l'étape 1 et ceux obtenus dans l'étape 2, et d'autre part les résultats de l'application d'une méthode de type des k-moyennes initialisée par une hiérarchie sur un corpus en langue arabe par rapport à ceux obtenus en langue française et anglaise. Nous pourrons, ainsi, confirmer ou infirmer qu'un Clustering par l'approche des k-moyennes hiérarchique pour un corpus de textes arabes, améliore les résultats d'une recherche documentaire et qu'une hiérarchie partielle peut être un bon point de départ pour les k-moyennes.

7 Références

1. [Abdelali et al, 2004] ; Ahmed Abdelali, Jim Cowie, Hamdy S. Soliman: «**Building A Modern Standard Arabic Corpus»,** New Mexico State University, 2004
2. [Ah-Pine et al, 2005] **;** Julien Ah-Pine, Julien Lemoine, Hamid Benhadda : **«Un nouvel outil de classification non supervisée de documents pour la découverte de connaissances et la détection de signaux faibles : rares text »,** île Rousse 2005, Journée sur les systèmes d'information élaborée
3. [Bellot et El-bèze, 2000] **;** Patrice Bellot, Marc El-Bèze: **«Clustering by means of Unsupervised Decision Trees or Hierarchical and K-means-like Algorithm »** RIAOÕ2000 Conference Proceedings - Collage de France, Paris, France, April 12-14, 2000, vol. I, pp. 344-363
4. [Bellot et El-bèze, 2000] **;** Patrice Bellot, Marc El-Bèze : **«Classification locale non supervisée pour la recherche documentaire** », T.A.L., 2000, vol . 41, n° 2, pp.
5. [Boulaknadel, 2005] **;** Siham Boulaknadel : «**Utilisation des syntagmes nominaux dans** un système de recherche d'information en langue arabe », LINA FRE CNRS 2729-Université de Nantes 2, France, 2005
6. [Chauché et al, 2003] **;** Jacques Chauché, Violaine Prince, Simon Jaillet, Maguelonne Teisseire : « **Classification automatique de textes à partir de leur analyse syntaxico-sémantique »** TALN 2003, Batz-sur-Mer, France
7. [Diab et al, 2004] **;** Mona Diab, Kadri Hacioglu, Daniel Jurafsky: «**Automatic Tagging of Arabic Text: From Raw Text to Base Phrase Chunks** », Stanford University, University of Colorado, Boulder, 2004
8. [Dunoyer, 2004] **;** Ludovic Denoyer : «**Apprentissage et inférence statistique dans les bases de documents structurés : Application aux corpus de documents textuels** », Université Paris 6, 2004
9. [Douzidia, 2004] **;** Fouad Soufiane Douzidia : «**Résumé automatique de texte arabe »,** Université de Montréal, 2004
10. [Fegas, 2005]; Mounir Fegas : « **Classification de documents XML, Application au** corpus d'INEX et aux rapports d'activité INRIA », LRI Université de Paris Sud XI
11. [Fortuna, 2004] **;** Blaž Fortuna, Marko Grobelnik, Dunja Mladeni : **«Visualization of text document corpus »,** JADT 2004 : 7es Journées internationales d'Analyse statistique des Données Textuelles
12. [Huot et al, 2005] **;** Charles Huot, Pascal Coupet : « Le Text Mining sur la langue Arabe : application au traitement des sources ouvertes », TEMIS SA, Paris, France, 2005
13. [Ignat et al, 2006] **;** Camelia Ignat, François Rousselot :« **Un algorithme de génération de profil de document et on évaluation dans le contexte de la classification thématique »,** JADT 2006 : 8es Journées internationales d'Analyse statistique des Données Textuelles
14. [Jaillet, 2003] **;** Simon Jaillet : « **Catégorisation automatique de documents** », LIRMM, Montpellier, France, 2003

15. **[**Jaillet et al, 2003**] ;** Simon Jaillet, Maguelonne Teisseire, Gérard Dray : «**Adéquation des modèles de représentation aux méthodes de catégorisation »**, LIRMM, Montpellier, France, 2003
16. **[**Jain et al, 2000**]** ; A.K. Jain, M.N. Murty, P.J. Flynn: « **Data Clustering: A Review »,** Michigan State University, 2000
17. **[**Jardino, 2004**] ;** Michèle Jardino : **« Recherche de structures latentes dans des partitions de textes de 2 à K classes »,** LIMSI – CNRS, France, 2004
18. **[**Larkey et al, 2002**] ;** Leah S. Larkey, Lisa Ballesteros, Margaret E. Connell: «**Improving Stemming for Arabic Information Retrieval: Light Stemming and Co-occurrence Analysis** », la 25ème édition de la conférence de la recherche et de développement en recherche d'information, (SIGIR 2002), Tampere, Finland, Août 2002
19. **[**Memmi, 2001**] ;** D. Memmi : **« Le modèle vectoriel pour le traitement de documents »,** 2001
20. **[**Turenne, 2000**] ;** Nicolas Turenne : **«** Apprentissage statistique pour l'extraction de concepts à partir du textes. Application au filtrage d'information textuelle **»,** Université Louis Pasteur, Starsbourg, 2000
21. **[**Réhel, 2005**] ;** Simon Réhel : **« Catégorisation automatique de textes et cooccurrence de mots provenant de documents non étiquetés »,** Faculté des études supérieures de l'Université Laval, QUÉBEC, 2005
22. **[**Sawaf et al, 2001**]** ; Hassan Sawaf, Jörg Zaplo et Hermann Ney: «**Statistical Classification Methods for Arabic News Articles** », AIXPLAIN AG Monnetstrasse 18 D-52146 Würselen, Germany, 2001
23. **[**Sulaiti, 2003**] ;** Latifa El Sulaiti : «**L**'arabe contemporain», Radio Qatar, 2003
24. **[**Weiss, 2006**]** ; Dawid Weiss: «**Descriptive Clustering as a Method for Exploring Text Collections**», Institute of Computing Science, Poznàn, Poland, 2006

Système de Classification Contextuel Appliqué à la Segmentation d'Images IRM

Abdelouahab Mousssaoui[1]

[1] Laboratoire Systèmes Intelligents (LSI), Université Ferhat Abbas de Sétif 19000, Algérie
moussaoui_abdel@yahoo.fr

Résumé. Le travail présenté dans cet article concerne l'application des méthode de fouille de données pour l'extraction puis la classification de données numériques. Nous avons développé à cette fin un système de classifieurs flou ne faisant aucune hypothèse quant à la distribution de la densité de probabilité (PDF). Nous avons utilisé la méthode de Parzen-Rosenblatt pour estimer la PDF puis via une heuristique de labellisation contextuelle baptisé relaxation markovienne nous avons pu repousser les limites de notre système de classifieurs. Afin de valider notre système de classification nous avons segmenté des images IRM de différentes pondérations. Une analyse qualitative et quantitative montre bien la robustesse de notre approche face aux données incomplète et/ou imprécise.

Mots-clés: Data mining, ECD, Classification, Fusion de données, PDF, Segmentation d'images, IRM

1 Introduction

Afin de reconnaître les différentes structures cérébrales et afin de détecter et de localiser les éventuelles anomalies, les médecins spécialistes utilisent, en plus des informations présentes dans les images, leurs propres connaissances et expériences afin de segmenter ces images [1][2]. L'analyse des données ainsi que les traitements qui en découlent dépendent de la qualité de cette segmentation qui constitue l'étape primordiale pour tout traitement ultérieur.
Devant cette masse extraordinaire de données médicales allant des données cliniques aux différents examens, aux images de différentes modalités (IRM, Scanner, TEP, etc.), les spécialistes n'utilisent en fait qu'une infime partie de ces données et les résultats obtenues ne correspondent pas, la plupart du temps, à la réalité.
Si les spécialistes arrivent tant bien que mal à classer les différentes données des images avec une précision plus ou mois acceptable, ils ne pourront jamais exprimer ni modéliser le processus de reconnaissance qu'ils ont adopté. Il est impératif d'utiliser les nouvelles technologies de traitement et d'analyse de données complexes afin d'extraire les informations pertinentes qui nous permettent, par la suite, de pousser les limites de nos connaissances. Ces nouveaux outils baptisés outils de data mining (ou fouille de données) nous permettent d'exploiter de grandes masses de données aussi complexes soient-elles, de fusionner différents types d'information,

d'extraire les connaissances et les règles d'associations à partir de ces données, etc. [3][4]. Les connaissances extraites seront injectées dans les algorithmes de segmentation et/ou de classification d'images sous formes de règles ou d'hypothèses implicites, ce qui conduit à des résultas meilleurs.

Plusieurs approche ont alors vues le jour à la fois contradictoires et complémentaires. Dans [5], Sonka et al. proposent une méthode de segmentation des tissus cérébraux à partir d'un atlas anatomique par regroupement de régions homogènes moyennent un paramètre de similarité utilisant la relaxation géométrique. D'autres chercheurs ont utilisé plutôt le gradient pour mesurer l'écart d'intensité entre les différentes structures issues de pondérations différentes, comme le cas de détecteur de Chenyang Xu [6][7]. D'autres méthodes plus constructives basées sur la coopération des deux approches régions et contours furent utilisés pour l'extraction des structures pathologiques à partir de données cliniques. L'algorithme le plus intéressent est sans doute celui de German et al. [8] qui appliqué à la détection des lésions cérébrales à travers un système de segmentation hybride basée sur des agents autonomes montre un taux de reconnaissances très appréciable.

Toutes ces techniques n'utilisent en fait qu'une partie très réduite de l'information par rapport à la masse importante des données, à la fois cliniques et anatomiques, qu'on peut avoir. Cela s'est répercuté sur la qualité des résultats obtenus et a même entraîné des prédictions erronées qui se manifestent souvent sous formes de mauvais diagnostiques ou de fausses alertes. Si la seconde prédiction erronée n'en cours pas vraiment de risques majeurs aux patients, on ne peut pas dire autant pour la première.

De nouvelles tendances ont alors émergées profitant de maximum de données disponibles et des nouvelles technologies pour l'extraction et l'intégration de connaissances. Il s'agit des techniques de segmentation par les outils de data mining. En effet, ces techniques exploitent le maximum de données possibles en faisant recours au techniques de data mining afin de repousser les limitent de la segmentation. Parmi ces nouvelles tendances on site les travaux de Babel et al. [9] qui furent les premiers à avoir intégré un procédé d'extraction de nouvelles connaissances dans leur algorithme de segmentation.

L'objectif de cet article est justement de montrer qu'il est possible de développer un cadre commun de fouille de données et de segmentation d'images IRM. Cette nouvelle approche est caractérisée par l'ajout de deux processus d'extraction de connaissance. Le premier est beaucoup plus exploratoire, il concerne essentiellement l'estimation de la densité de probabilité (PDF) quant au second il est beaucoup plus contextuelle et il concerne la labellisation contextuelle des voxels à partir d'un atlas de référence [10]. Dans la deuxième section nous décrivons en détail le processus d'estimation de la PDF par la méthode de Parzen-Rosenblatt. La section 3 quant à elle est consacrée à la labellisation contextuelle via la relaxation markovienne. La section 4 décrit en détail l'approche proposée. Les tests et les résultats obtenus font l'objet de la section 5.

2 Estimation de la densité de probabilité (PDF)

2.1 Définition de la PDF

Considérons $\mathbb{R}^n$ un espace de dimension n . Considérons un échantillon de N données $(X_i)_{1\le i\le N}$ représentées par N points avec :

$X_i = \left(x_{i,1},\ x_{i,2},\ ...,\ x_{i,n}\right)$

La fonction de densité de probabilité est définie sur $\mathbb{R}^n$ par une fonction ϕ satisfaisant au moins aux conditions suivantes :

$$\begin{cases} \forall x \in \mathbb{R}^n, \phi(x) \ge 0 \\ et \\ \int_{\mathbb{R}^n} \phi(x)\ dx = 1 \end{cases} \quad (1)$$

Dans la pratique, cette fonction ϕ est inconnue, il existe plusieurs méthodes pour l'estimer.

La méthode la plus répandue consiste à approcher la fonction ϕ par un histogramme, mais l'estimation obtenue ne présente pas les propriétés de continuité requises par la plupart des applications. De plus, les largeurs des classes (ou éléments) de l'histogramme sont délicates à déterminer.

Parmi les méthodes d'estimation de la densité par une fonction continue, lune des plus connues est la méthode des k plus proches voisins. L'estimations de la fonction de densité est définie par :

$$\hat{\phi}(x) = \frac{k-1}{2Nd_k(x)} \quad (2)$$

où k est le nombre fixé des plus proches voisins (typiquement $k = \sqrt{N}$ d'après Silverman [11] et $d_k(x)$ est la distance de x à son $k^{ième}$ plus proche voisin (en classant par ordre croissant les distances de x à chacune des N données, $d_k(x)$ est la $k^{ième}$ distance).

L'estimation par la méthode des k plus proches voisins ne permet pas d'obtenir une fonction dérivable et de plus elle donnera des pics correspondant à des donnés situés dans les régions les plus denses. Dans la littérature, on considère que ces estimations continues mais non dérivables de la densité ne sont pas assez lissées et restent assez approximatives [12][13].

Les méthodes basées uniquement sur le maximum de vraisemblance permettent de dériver la fonction estimée, mais l'interpolation entre les données est purement heuristique. La vraisemblance de l'estimée $\hat{\phi}$ de la densité pour l'échantillon $(X_i)_{1\le i\le N}$ est définie par :

$$L\left(\hat{\phi}/\left(X_i\right)_{1\le i\le N}\right)=\prod_{i=1}^{N}\hat{\phi}\left(x_i\right) \tag{3}$$

Maximiser la vraisemblance ne permet pas d'estimer la densité en dehors de l'échantillon $(X_i)_{1\le i\le N}$ (quid de $\hat{\phi}(x)$ pour $x\neq X_i$). Il faut imposer des restrictions à $\hat{\phi}$ permettant une interpolation acceptable de $\hat{\phi}$ entre les données. Ces restrictions ou pénalités ont pour but de lisser l'estimation de la fonction. En dimension 1, on utilise le plus souvent un critère du type suivant :

$$\ell_\alpha\left(\phi\right)=\sum_{i=1}^{N}\log f\left(x_i\right)-\alpha\int_{-\infty}^{\infty}\left(\phi''\right)^2 \tag{4}$$

Le premier terme $\sum_{i=1}^{N}\log\ \phi\left(x_i\right)$ est le logarithme de la vraisemblance qu'il faut maximiser, le deuxième terme $\alpha\int_{-\infty}^{\infty}\left(\phi''\right)^2$ est une pénalité qu'il faut minimiser et qui est contrôlée par le paramètre α. Ce paramètre est ajusté pour lisser plus ou moins l'estimation de la densité [12]. En dimension 2 ou plus, cette méthode des pénalités est plus délicate à mettre en place, il faut trouver le bon compromis entre le paramètre de lissage et la vraisemblance. Dans ce type d'approche, la part d'heuristique est importante. A cause de ce manque de rigueur, il nous a semblé préférable d'estimer la densité par la classique méthode du noyau (encore appelée méthode Parzen-Rosenblatt) décrite ci-après.

2.1 Fenêtres de Parzen

La probabilité P qu'un vecteur x de l'espace $\mathbb{R}^n$ tombe dans une région H de $\mathbb{R}^n$ s'exprime avec la densité de probabilité p dans la région H :

$$p\left(x\in H\right)=\int_H p\left(x\right)dx \tag{5}$$

Si la région H est suffisamment petite et si p est continue, une approximation de cette probabilité s'écrit :

$$p\left(x\in H\right)=p\left(x\right)V\left(H\right) \tag{6}$$

avec x vecteur de H, et $V\left(H\right)$ volume de H.

Pour estimer la densité de probabilité en x, formons une suite de régions H_N contenant x, de volume $V\left(H\right)$ et correspondant à N tirages dans $\mathbb{R}^n$. Les estimations successives valent donc :

$$p_N\left(x\right)=\frac{K_N}{N.V\left(H_N\right)} \tag{7}$$

Les fenêtres de Parzen font partie des méthodes qui permettent de construire la suite de fonctions de densité de probabilité $x\rightarrow P_N\left(x\right)$ où

Considérons la fonction fenêtre centrée en 0 :

$$\phi(u) = \begin{cases} 1 & |u| \le \frac{1}{2} \\ 1 & \sin on \end{cases} \tag{8}$$

Prenons pour régions H_N autour de x l'hypercube de côté de largeur h_N et entré en x ; son volume est $V(H_N) = h_N^p$; le nombre K_N de vecteurs x_l issus de N tirages dans E et tombant dans H_N s'exprime alors suivant :

$$K_N = \sum_{i=1}^{N} \phi\left(\frac{x - x_l}{h_N}\right) \tag{9}$$

En reportant cette expression dans (7), on obtient :

$$p_N(x) = \frac{\sum_{l=1}^{N} \phi\left(\frac{x - x_l}{h_N}\right)}{N.h_N^p} \tag{10}$$

Dans un souci de généralité, nous ne nous restreindrons pas à une fonction fenêtre mais à une fonction densité vérifiant l'équation (1).

3 Labellisation contextuelle

Les méthodes non-contextuelles de classification considèrent que l'appartenance d'un voxel à une classe ne dépend que de la radiométrie du voxel; aussi, les segmentations produites ne respectent pas toujours la cohérence spatiale des structures [14]. Du fait du bruit et de la présence de volumes partiels, la loi statistique caractérisant la radiométrie d'une classe recouvre presque systématiquement la loi d'une ou de plusieurs autres classes. L'application de seuils radiométriques pour décider de l'appartenance d'un voxel à une classe cause donc des erreurs, qui pourraient être évitées si des connaissances portées par le voisinage de ce voxel étaient prises en compte. Dans notre travail, nous avons opté pour la relaxation markovienne dont les détails seront décrits dans la section suivante et pour laquelle nous avons pris deux modèles de régularisation topologique et bayésiènne.

3.1 Champs de Markov

Les champs de Markov permettent de modéliser des propriétés globales en utilisant des contraintes locales.

Nous notons S $(card(S) = N)$ le réseau des sites de l'image et $V = \{V_s / s \in S\}$ le système de voisinage (V_S est le voisinage du site s).

Nous notons $X = \{X_1, ..., X_N\}$ une famille de variables aléatoires définies sur le réseau S. Chaque variable aléatoire est à valeur dans Ω. Nous notons $X_i = x_i$ pour

signifier que la variable aléatoire X_i prend la valeur x_i $(x_i \in \Omega)$. De même nous notons $X = x, x \in \Omega^N$, pour signifier l'événement suivant : $(X_1 = x_1, ..., X_N = x_n)$.
x, appelé configuration, correspond à une réalisation du champ aléatoire X. Cette configuration a une certaine probabilité que nous notons $p(X = x)$. Faisons l'hypothèse que la distribution des intensités dans chaque classe, suit une loi normale dont les paramètres moyenne m_λ et variance σ_λ^2.
X est appelé champ de Markov par rapport au système de voisinage V s'il vérifie les deux propriétés suivantes :

$$1- \quad \forall x \in \Omega^N, p(X = x) > 0$$

$$2- \quad \begin{cases} - \ \forall s \in S \ \ et \ \ \forall x \in \Omega^N, \\ P\left(X_s = x_s / X_r = x_r, r \in S - \{s\}\right) = P(X_s = x_s / X_r = x_r, r \in V_s) \end{cases} \tag{11}$$

Dans la suite, pour faciliter les notations, nous confondons X et x. $P(X = x)$ est alors notée $p(x)$.

3.2 Modèle de Régularisation

3.2.1 Cadre topologique

Considérons l'image comme un ensemble de sites S. Chaque site s de S correspond à une localisation particulière dans l'image : ce peut être un voxel ou une région de l'image (dans ce dernier cas, les régions sont disjointes deux à deux).
Ajoutons une dimension topologique au problème en donnant à S une structure de graphe grâce à la définition pour chaque site s de son voisinage :
$V_s = \{s' \in S, \ s' \neq s \ et \ s \in V_{s'}\}$. Ce graphe est un réseau régulier si S est l'ensemble des voxels d'une image, ou un graphe d'adjacence si S est un partitionnement de l'image en régions.
A la théorie des graphes nous empruntons la notion de clique, configuration de sites tous voisins deux à deux, ou le site lui-même.
Nous appelons clique générique un type de configuration[1]; une clique est alors une instance particulière d'une clique générique.
Prenons deux exemples. La configuration qui correspond au site unique est une clique générique; une instance de cette dernière est une clique qui ne comprend qu'un site donné du graphe. La configuration qui est définie sur deux sites d'un réseau régulier 3D, tel que l'un d'eux a pour coordonnées relatives par rapport à l'autre (+0, -1, +1) dans ce réseau, est une seconde clique générique; une instance de cette dernière est par exemple la clique {(3, 3, 3), (3, 2, 4)}.

[1] Nous avons introduit cette notation, originale à notre connaissance, dans le but de formaliser avec un peu plus de précision certaines expressions.

L'ordre o_C d'une clique générique c est le nombre de sites dans la configuration qu'elle d´écrit.
Nous noterons C l'ensemble des cliques génériques définies sur le graphe. Une clique générique $c \in C$ se traduit généralement par plusieurs cliques sur le graphe S ; notons l_c ce nombre de cliques :

$$c \longrightarrow c^1,\ldots,\ c^{l_c} \quad avec \quad c^l = \left\{ s^1\left(c^1\right),\ldots,\ s^{o_c}\left(c^l\right) \right\} \tag{12}$$

où $s^q\left(c^l\right)$ est le $q^{ème}$ site de la clique c^l . Classiquement, l'ordre d'une clique c^l est le nombre de sites qui la composent; c'est donc o_c .
Une constatation importante est qu'un même site peut intervenir dans plusieurs cliques, ce qui s'écrit :

$$\left(l \neq l'\right) \ \Rightarrow \ \left(\forall q \ \forall q' \ s^q\left(c^l\right) \neq s^{q'}\left(c^{l'}\right)\right) \tag{13}$$

3.2.2 Cadre Bayésien

Effectuer une classification bayésienne revient à maximiser la probabilité a posteriori (MAP en abrégé) : $p\left(\Upsilon = Y^{(t)} / \chi = X\right) = \dfrac{p\left(\chi = X / \Upsilon = Y^{(t)}\right) p\left(\Upsilon = Y^{(t)}\right)}{p\left(\chi = X\right)}$ (14)

or $p\left(\Upsilon = Y^{(t)} / \chi = X\right) \propto p\left(\chi = X / \Upsilon = Y^{(t)}\right) p\left(\Upsilon = Y^{(t)}\right)$, où $\propto$ signifie "proportionnel à".
Sous l'hypothèse que les observations en chaque site sont indépendantes les unes des autres, ce qui suppose implicitement que le bruit soit lui aussi indépendant d'un site à l'autre, nous pouvons écrire : $p\left(\chi = X / \Upsilon = Y^{(t)}\right) = \prod_{s \in S} p\left(x_s = x_s^I / \Upsilon = Y^{(t)}\right)$ (15)

Cette hypothèse est raisonnable car, d'une part les images par résonance magnétique sont acquises puis reconstruites de telle façon que les données radiométriques obtenues pour un voxel ne sont pas (en première approximation) influencées par la proximité d'autres voxels, et d'autre part le bruit est considéré comme additif en chaque voxel et supposé gaussien.
De plus, grâce à l'indépendance spatiale du bruit, la radiométrie en un site ne dépend que de la classe en ce site, d'où finalement : $p\left(\chi = X / \Upsilon = Y^{(t)}\right) = \prod_{s \in S} p\left(x_s = x_s^I / y_s = y_s^{(t)}\right)$ (16)

Enfin, nous supposons que la densité de probabilité conditionnelle $p\left(x_x = x_s^I / y_s = y_s^{(t)}\right)$ suit une loi gaussienne, ce que nous avons justifié dans la section précédente.
Pour des facilités d'écriture qui apparaîtront par la suite, introduisons la fonction U^a vérifiant :

$$U^a\left(x_s^I, y_s^{(t)}\right) = -\log\left(p\left(x_s = x_s^I / y_s = y_s^{(t)}\right)\right) \quad (17)$$

Remarquons qu'elle n'est définie que si la densité de probabilité conditionnelle n'est jamais nulle.

Dans le cas de données monodimensionnelles, U^a s'écrit :

$$U^a\left(x_s^I, y_s^{(t)}\right) = \frac{\left(x_s^I - v_{is}(t)\right)^2}{2\sigma_{is(t)}} + \log\left(\sqrt{2\pi}\sigma_{is(t)}\right) \quad (18)$$

Afin de mener à bien la classification, il nous reste encore à déterminer la loi de probabilité globale $p\left(\Upsilon = Y^{(t)}\right)$, qui malheureusement ne peut être calculée directement.

4 Approche proposée

Dans cette section nous allons présenter l'algorithme de segmentation par approche de classification. L'algorithme est composé de quatre phases : l'estimation de la densité de probabilité (PDF) par la méthode de Parzen-Rozenblatt qui va nous renseigner sur le nombre de classes à priori, puis une première classification aveugle est opérée utilisant le nombre de PDF comme seule connaissance, la troisième phase consiste à faire une labellisation contextuelle par une approche markovienne moyennent un atlas anatomique ; quant à la quatrième et dernière phase elle consiste à segmenter les images à partir des classes obtenues.

Algorithme :

- étape 1: (Estimation de la PDF)
  - Fixer le noyer K , fixer la taille de la fenêtre h
  - $\forall x \in \mathbb{R}^n$, calculer $\hat{\phi}(x)$ l'estimation de $\phi(x)$ par la formule (2).
- étape 2: ($1^{ère}$ classification utilisant le nombre de PDF comme seule connaissance)

  - Minimiser l'erreur quadratique moyenne $EQMI = E\left(\int_{R^d}\left(\hat{\phi}(x) - \phi(x)\right)^2 dx\right)$

  Sous la contraint de minimisation $K_E(x) = \begin{cases} \frac{1}{2} C_d^{-1}(d+2)\left(1 - \|x\|^2\right) & si \|x\| < 1 \\ 0 & \text{sinon} \end{cases}$

  où C représente le nombre de PDF obtenue à partir de la phase 1.
- étape 3: ($2^{ème}$ classification par labellisation contextuelle)

  - Maximiser les probabilités marginales a posteriori (MPM) selon la formule (13) ou (14) avec comme conditions les contraintes décrites respectivement par les formules (12) ou (16).
- étape 4: (Segmentation de l'image)
  - Segmenter l'image à partir des données classées

5 Résultats

Afin de valider notre approche, nous avons tout d'abord testé notre algorithme sur de données simulées comportant 675 points représentants 4 classes différentes, nous avons estimés la densités de probabilités de ces points selon trois types de noyau (Gaussien, Aplati et le noyau d'Epanechnikov) [4][15], les résultats sont très intéressant parce qu'on arrive déjà à partir de l'estimation de la densité de probabilité d'avoir une connaissance qui représente le nombre de classes (voir Fig. 1. et Fig. 2.). Cette connaissance sera injectée dans le processus de classification. Une labellisation contextuelle corrige les erreurs résiduelles et augmente le taux de reconnaissances des points. Un taux de recouvrement de plus de 98 % a été atteint avec des données simulées (voit Tab. 1.). Cependant il faut noté que le noyau d'Epanechnikov donne une meilleure estimation de la PDF et par la suite la meilleurs classification.
Une deuxième série de test a été faite sur des images de type IRM, pour les quels nous avons appliqué notre algorithme afin de reconnaître les différentes structures cérébrales. Les résultats sont présentés sur le tableau 1 où nous pourrons constaté que effectivement les connaissances incluses dans le processus de segmentation augmentent le taux de bonnes classifications et que a relaxation markovienne pousse encore les limites de la classification. Cela se traduit dans le processus de segmentation par une bonne séparation des principales substances cérébrales (MB, MG, LCR, Autre). Une nette séparation apparaît dans la coupe sagittale car effectivement pour l'atlas utilisé la coupe sagittale est mieux représentée et les parties de l'anatomie cérébrale sont bien détachées par rapport au deux autres coupes (voir Fig. 3.).
Afin de valider les résultats et avons d'avoir une comparaison objective, nous avons utilisé deux métriques de mesures de similarité une quantitative et l'autre qualitative, il s'agit respectivement de la mesure de similarité kappa et du taux de recouvrement souvent appelé la mesure de similarité de Vinet [16]. Les résultats sont très intéressants et ils sont représentés dans le tableau 2.

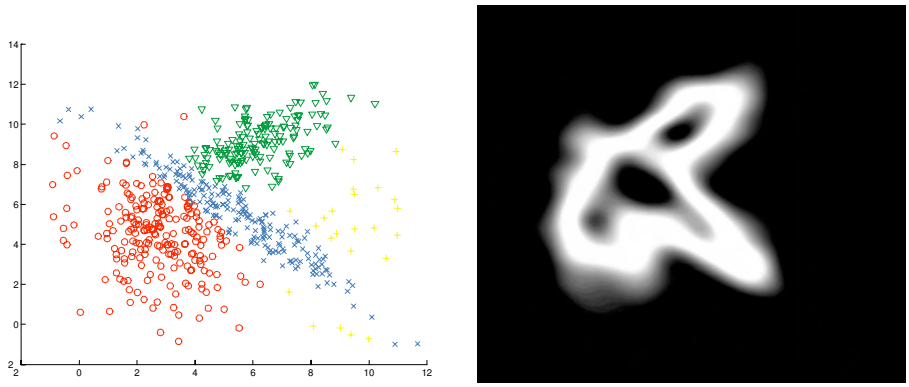


Fig. 1. Premier ensemble de simulation et l'estimation des pdf (noyau gaussien)

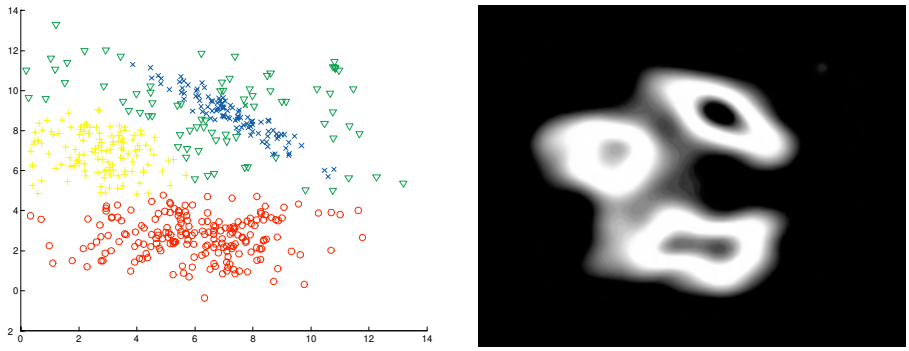


Fig. 2. Deuxième ensemble de simulation et l'estimation des PDF (noyau d' d'Epanechnikov)

%	Classification	Relaxation (cadre topologique)	Relaxation (cadre Bayésien)
MB	87	90	98
MG	91	78	94
LCR	89	89	98
Autres	91	92	97

Tab. 1. Pourcentage de recouvrement des différents tissus (MB, MG, LCR et autres) après la labellisation contextuelle.

Kappa T.R	Coupes	Classification	Relaxation (cadre topologique)	Relaxation (cadre Bayésien)
	Sagittales	0.85 98.01	0.90 95.43	0.90 95.43
	Coronales	0.63 96.99	0.68 97.39	0.75 97.39
	Axiales	0.49 95.0	0.86 94.39	0.73 94.39

Tab. 2. Mesures de la segmentation selon les deux directions sagittale et coronale : utilisation de la mesure de Kappa et du taux de recouvrement (%).

(a) **(b)** **(c)**

Fig. 3. (a) image originale, (b) segmentation après relaxation topologique, (b) segmentation après relaxation probabiliste

6 Conclusion

En conclusion, on peut dire que nos algorithmes de segmentation de l'encéphale accompagné des heuristiques de labellisation contextuelle ou de coopération sont très robustes. Les résultas sont très convenables si on compare les segmentations automatiques avec celles fournies par les experts. Cet outil d'imagerie neuroanatomique nous a permis de segmenter avec une grande précision sur des volumes de l'encéphale humain avec un taux de reconnaissance moyen de 98 % par rapport aux structures identifiées par les experts. De plus, la qualité de la segmentation est en faveur des méthodes utilisant des connaissances supplémentaires (topologique ou probabiliste).
Nous avons montré l'efficacité de ces algorithmes de segmentation selon les différentes directions. En plus la fusion de données à partir de plusieurs coupes complémentaires semble être très satisfaisante quant à la qualité de la segmentation. La sensibilité d'une segmentation par rapport à une autre est due à des connaissances psychologiques implicites propres à chaque expert (orientation dans l'espace, projection mentale, extrapolation, etc.) qui ne sont pas pris en compte par nos méthodes. Il serait peut être indispensable de développer un système expert avec une base de connaissance très développée pour remédier à ce type de problème.

7 Bibliographie

1. Szilagyi, L.; Benyo, Z.; Szilagyi, S.M.; Adam, H.S.; "MR brain image segmentation using an enhanced fuzzy C-means algorithm", Engineering in Medicine and Biology Society, Proceedings of the 25th Annual International Conference of the IEEE , Volume: 1 , 17-21 Sept. 2003.

2. Shi Juan He; Xia Weng; Yamei Yang; Weili Yan; "MRI brain images segmentation", Circuits and Systems, The 2000 IEEE Asia-Pacific Conference on , 4-6 Dec. 2000,pp. 113-116.
3. A.W.-C Liew,; Hong Yan, "Adaptive fuzzy segmentation of 3D MR brain images", *The 12th IEEE International Conference on Fuzzy Systems*, May 2003, vol. 2, pp. 25-28.
4. A. Moussaoui, K. Benmahammed and V. Chen, "A New Fuzzy Clustering Based Mean-Shift Procedure: Applied to brain image segmentation", *Asian Journal of Information Technology* (AJIT), 2005, vol. 4, no. 6, pp. 590-595.
5. M. Sonka, S. Tadikonda, and A. M.Collins. "Knowledge-Based Interpretation of MR brain Images". IEEE ransactions on Medical Imaging, 15(4), 1996, pp. 126-137.
6. Xu Chenyang, "Deformable models with application to human cerebral cortex reconstruction from magnetic resonance images", PhD thesis. University of Baltimore, Maryland, USA, April, 2000.
7. Xu Chenyang, "Image Segmentation Using Deformable Models", IEEE transactions on Medical Imaging, 12(4), 2003, pp. 252-260.
8. L. Germond, M. Dojat, C. Taylor, and C. Garbay. "A multi-agent system for MRI brain segmentation". In 7th European Conference on Artificial Intelligence and Medecine, 423–432. Lecture Notes in Artificial Intelligence 1620, Springer, 1999.
9. L.K. Babel, C. Patric, "Image mining for human cerebral detection", IEEE transactions on Medical Imaging, 10(3), 2005, pp. 121-147.
10. A. Moussaoui and V. Chen. : Segmentation d'images médicales par coopération de classifieurs. In JETIM, pp. 57-62, Novembre (2004).
11. J.C. Bezdek and N.R. Pal. "Some new indexes of cluster validity". IEEE Trans. Systems Man. Cybern., 28 : 301-315, 199.
12. B. W. Silverman. "Density Estimation for Statistics and Data Analysis". Chapman and Hall, London, 1986.
13. L. Devroye. "A Course in Density Estimation". Volume 14 of Progress in Probability and Statistic. M. Rosenblatt, Birkhauser, Boston, 1987.
14. Te-Won Lee; Lewicki, M.S.; "Unsupervised image classification, segmentation, and enhancement using ICA mixture models", Image Processing, IEEE Transactions on , Volume: 11 Issue: 3 , March 2002, pp. 270 -279.
15. Y. Cheng, Mean Shift, Mode Seeking and Clustering, IEEE Trans. Pattern Analysis and Machine Vision, vol. 17(8), August 1995, pp.790-799.
16. L. Vinnet, "Segmentation et mise en correspondance de régions de paires d'images stéréoscopiques, " Université de Paris IX - Dauphine, 1991.

Optimisation Multi Critères par Algorithmes Génétiques du Design des Réseaux d'Osmose Inverse

Mehdi Metaiche [1] et John Palmeri [2]
1 Institut Europeen des Membranes-CNRS, Montpellier
Email: metaiche@iemm.univ-montp2.fr
2 Laboratoire de Physique Théorique, Université Paul Sabatier, Toulouse

Résumé:
L'objectif de ce travail est l'optimisation de la conception des réseaux d'osmose inverse, dans le but de les utiliser dans des installations industrielles de dessalement de l'eau de mer. Une modélisation à plusieurs échelles est effectuée : échelle de la membrane (physique), échelle du réseau (conception), et échelle des coûts (économique).Le travail d'optimisation consiste à concevoir et exploiter un réseau d'osmose inverse capable de produire de l'eau –eau de mer dessalée- à faible prix tout en respectant l'exigence sur la salinité totale et la teneur en Bore (élément très toxique).Vu la non linéarité et la complexité de la fonction objectif, ainsi que la multitude des contraintes et leur non linéarité, un algorithme évolutionnaire de type génétique est utilisé, doté d'une sélection par la roue biaisée de Goldberg avec un Scaling exponentiel à coefficient cinq, dont une étude de convergence est réalisée.
Le travail a conduit à l'élaboration d'un code d'optimisation (logiciel) : DESALTOP par le biais de la programmation orienté objet sous Compaq Visual Fortran, permettant d'optimiser les conceptions, et les différents paramètres de fonctionnement des réseaux d'osmose inverse avec le contrôle de la salinité totale et de Bore, dont l'étude large, menée pour la validation a démontré l'efficacité.

Mots clés :
Conception, dessalement, eau, optimisation, osmose inverse, réseaux.

1- Introduction:

L'eau est une ressource vitale, précieuse et rare. Son rôle économique et social est très important. Et comme les ressources naturelles (conventionnelles) en eau douce sont limitées (et épuisées dans certaines régions), le recours aux sources non conventionnelles est primordial. Dans cette optique, le dessalement de l'eau de mer et des eaux saumâtres constitue une solution prometteuse au problème de l'insuffisance de ces ressources naturelles.

Selon le « 2004 IDA Worldwide Desalination Plants Inventary Repor », au début de l'année 2005, il y avait plus de 10.350 unités de dessalement dans

plus de 140 pays du monde, qui produisent plus de 37.750 Millions m^3/j de l'eau dessalée, adaptée à la consommation courante. La technique de dessalement peut traiter les eaux de différentes origines : l'eau de mer (dont la concentration en sels varie entre 35.000 à 49.000 ppm), l'eau saumâtre et l'eau souterraine (ou la concentration varie de 1.000 à 10.000 ppm), l'eau de drainage, l'eau usée, l'eau polluée par les métaux lourds, et l'eau polluée par les radioactifs. La technique de dessalement par osmose inverse représente un choix plus raisonnable et un plus économique par rapport aux techniques non-membranaires. D'après le même rapport, plus de 40% de la production mondiale en matière de dessalement provient de l'osmose inverse. Le développement technique des systèmes d'osmose inverse peut conduire à (i) améliorer leurs performances, (ii) trouver des conceptions plus rentables, (iii) produire de l'eau de bonne qualité et à faible prix, (iv) réduire de façon importante les coûts, et (v) satisfaire les besoins en eau de plus en plus importants.

Le but de ce travail est l'optimisation multi-critères de systèmes d'osmose inverse par :
- La modélisation des transferts à travers les membranes d'osmose inverse.
- La modélisation de la conception des procédés de dessalement de l'eau par membrane d'osmose inverse.
- La modélisation des différents coûts.
- Et l'élaboration d'un code de calcul pour la conception optimisée des ces réseaux, la prédiction de la qualité de l'eau dessalée, le calcul des coûts et du prix de revient unitaire, et le contrôle des teneurs en Bore dans l'eau produite.

La modélisation de transport à travers les membranes d'osmose inverse et dans les modules attire de plus en plus l'attention des chercheurs avec l'objectif de fournir des modèles fiables pour la prédiction des performances [1-11]. La modélisation des performances des installations pour le dessalement de l'eau a également fait le sujet de nombreuses études [12-17]. La recherche des conceptions nouvelles reste de l'actualité avec le but de créer de nouveaux design encore plus performants [18-22].
Il existe une multitude de méthodes et d'algorithmes d'optimisation. Les algorithmes à gradient convergent vers l'optimum si la fonction objectif est convexe, sinon leur convergence est locale [23]. Les méthodes de programmation linéaire PL traitent des problèmes dont la fonction objectif est continue et les contraintes sont continues [24]. La méthode du simplexe est la plus répandue pour la résolution d'un tel problème [25] ; les étapes de la méthode repose sur la recherche d'une solution possible pour démarrer les itérations. Les méthodes non linéaires sont utilisées pour les problèmes non-linéaires à variables discrètes [26].

Les algorithmes d'optimisation stochastiques intègrent des mécanismes aléatoires d'exploration de l'espace de recherche. Ces mécanismes interviennent à différentes étapes de l'exploration suivant l'algorithme utilisé. On peut dire en première approximation, qu'ils permettent avec une certaine probabilité de s'échapper des minima locaux [24] (voir la figure 1 qui présente une fonction objectif non-linéaire avec plusieurs optimums locaux dans l'espace des paramètres). Les principales caractéristiques de ces méthodes stochastiques [24] qui sont inspirées de processus naturels sont l'exploration de l'espace des paramètres selon une composante aléatoire, un temps de calcul long, et une robustesse qui permet de les appliquer sans connaître a priori les propriétés du domaine de recherche [27].

Les algorithmes stochastiques les plus célèbres sont : L'algorithme de recuit simulé, l'algorithme de la colonie de fourmis et les algorithmes évolutionnaires.
Les algorithmes évolutionnaires (AE) (1965) sont inspirés du paradigme de l'évolution Darwinienne des populations, où l'évolution se traduit par un processus itératif de recherche de l'optimum dans l'espace de décision (de recherche). La performance des solutions étant évaluée par les valeurs de la fonction objectif [28]. Leur avantage crucial repose sur la possibilité de connaître les valeurs de la fonction objectif sans faire appel à sa dérivée et sans exiger son expression analytique. Les principaux AE sont: les algorithmes génétiques qui sont les plus connus et les plus utilisés [28], les stratégies d'évolution, la programmation évolutionnaire, et la programmation génétique.

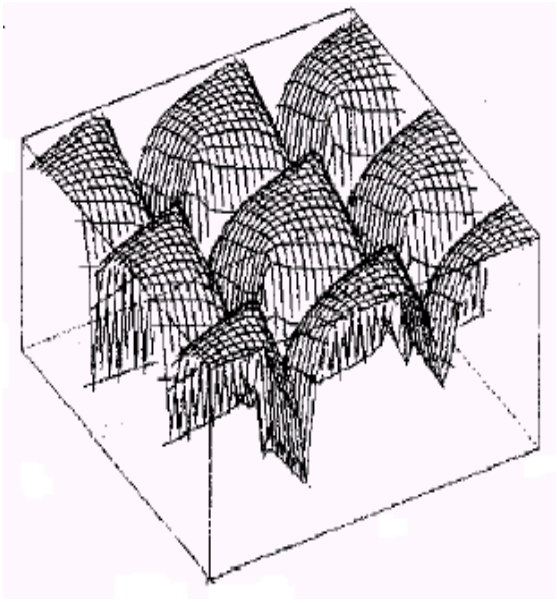


Figure 1: Fonction à plusieurs optimums locaux.

L'enchaînement des étapes d'un algorithme génétique consiste à (1) créer une population initiale (en exploitant la relation entre le nombre de variables dépendantes et indépendantes représentant le problème à résoudre [29]), (2) coder les individus (le codage permet d'effectuer les opérations génétiques [30], le choix du codage est une étape déterministe dans la conception d'un algorithme génétique [31]), (3) évaluer les individus et sélectionner les meilleurs qui constitueront la génération intermédiaire [29], (4) effectuer les opérations génétiques de croisement et de mutation [29,32,22], puis (5) l'évaluation des nouveaux individus et la sélection des meilleurs, et ainsi de suite, jusqu'à l'arrêt d'évolution de la fonction d'évaluation [33]. On peut améliorer la sélection vis-à-vis de la sensibilité aux écarts de la fonction d'évaluation, par d'autres modes (ranking) et d'autres principes (scaling, sharing) qui empêchent les individus "forts" d'éliminer complètement les plus "faibles" [34,32].

Les contributions pour optimiser les systèmes d'osmose inverse sont nombreuses, mais la quasi-totalité sont des travaux d'optimisation des paramètres de fonctionnement, en utilisant des méthodes analytiques classiques dont l'efficacité est limitée [17,24-25,29,33,35]. Certains travaux récents ont employé la modélisation non physique (réseaux neurones) pour optimiser ces paramètres au niveau d'une conception simple [32]. Chandan Guria and al. en 2005 [25], ont réussi à traiter une optimisation multi critère avec un algorithme évolutionnaire, mais pour une configuration du réseau très simple, sans arriver à toucher le problème des conceptions. Mais aborder des travaux d'optimisation multi critères des conceptions, basée sur une modélisation physique, nécessite la formulation d'une modélisation conceptuelle et la recherche d'un algorithme fiable.

2- Moyens et méthodes :

Le programme de travail consiste à réaliser trois parties principales: la modélisation des performances, l'optimisation multi-critères des systèmes (réseaux), et l'élaboration d'un code de calcul et de conception fiable.

2-1- Modélisation des performances, qui se fait à trois échelles différentes :

2-1-1- Echelle de la membrane d'osmose inverse (échelle physique) par la modélisation: de l'écoulement et du transfert des sels à travers la membrane (Kimura-Sourirajan), du transfert de Bore (Taniguchi), de la pression osmotique (Van't Hoff), du coefficient de transfert de masse (Sherwood, Reynolds and Schmidt numbers), de la polarisation (Film theory), de la conservation des masses, des pertes de charges (Hagen-Poiseuille and Ergun), et de la densité de flux, de la viscosité et la diffusion des sels (Sekino). La Figure 2 présente la corrélation entre le modèle et les douze mesures expérimentales du taux de rejet des sels obtenus par H.Ohya and al. [36] pour des membranes de type B9.

2-1-2- Echelle du réseau (échelle conceptuelle) : par la sélection de la gamme des conceptions efficaces, et la modélisation de leur comportement (flux, teneur en sels, teneur en Bore, consommation de l'énergie. Cela nécessite de modéliser les réseaux de membranes en série rejet, les réseaux en série production, et les réseaux hybrides rejet- production avec/sans bypass et mélange.
2-1-3- Echelle des coûts (échelle économique), ce qui consiste à modéliser les différents coûts de dessalement de l'eau, ainsi que le prix de revient unitaire final.

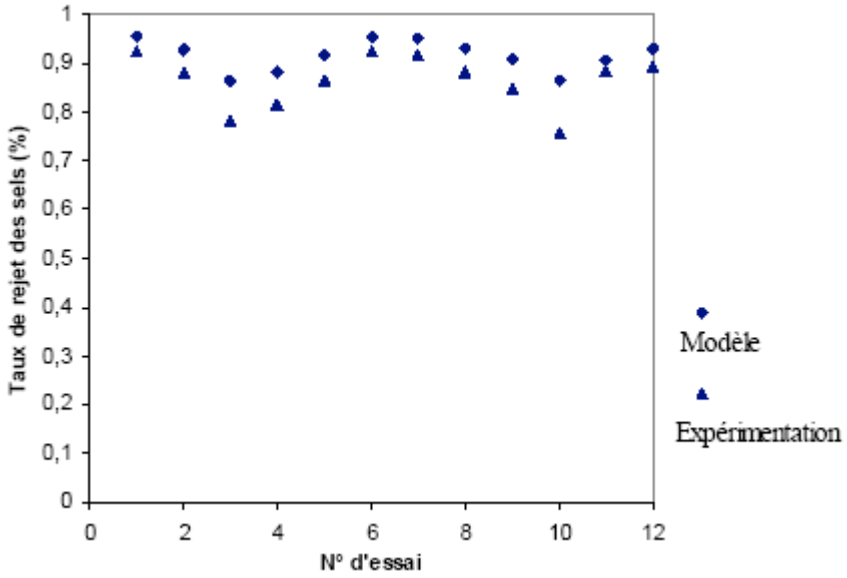


Figure 2: Corrélation des résultas du modèle avec l'expérimental.

2-2- Optimisation des procèdes d'osmose inverse, ce qui consiste à:
- Etablir les fonction objectifs: minimiser le prix unitaire de l'eau dessalé, minimiser la salinité totale, et minimiser la teneur en Bore dans le flux (le débit) produit.
- Etablir l'ensemble des contraintes qui gouvernent le problème à optimiser: la pression exercée sur les membranes dans les différents étages doit être supérieure à la pression osmotique résultante, et inférieur à la pression limite à partir de laquelle résiste la membrane. Et le flux d'alimentation du système doit être positif, et inférieure à une valeur limite à partir du quelle, la pression osmotique devient négative et le transport à travers la membrane devient impossible.
- Et adapter les algorithmes évolutionnaires pour le traitement du problème: nombre d'individu dans la population initiale, choix de codage, choix des types de croisement et de mutation, choix de leurs probabilité, etc.

2-3- Elaboration du code de calcul, ce qui consiste en la réalisation d'un produit informatique (logiciel) capable de concevoir des systèmes d'osmose inverse optimaux (plus efficaces et moins coûteux), de choisir les paramètres de fonctionnement optimaux et d'effectuer les calculs de prédiction des flux, des qualités et des coûts.
L'étude de fiabilité du code (modèle) est à réaliser à trois niveaux :
- la validation de la fiabilité pour le calcul des performances des membranes.
- l'étude de convergence de l'algorithme évolutionnaire choisi.
- l'étude de validation globale et d'utilité du code.

Nous avons opté pour les algorithmes génétiques comme méthode d'optimisation, grâce à leur efficacité pour traiter les problèmes à fonctions objectifs non-linéaires et à plusieurs contraintes non-linéaires.
La programmation orienté objet avec Compaq Visual Fortran a été adoptée afin d'automatiser les calculs et mettre en ouvre un logiciel de calcul et de conception optimisés.
En plus, vu que la toxicité de Bore (molécule très fine) constitue un vrai problème pour les membranes d'osmose inverse, nous avons consacré un travail important à cette question en modélisant son taux de passage à l'échelle de la membrane, ainsi qu'à l'échelle d'un système, par l'ajout d'une contrainte réservée à son contrôle, et par l'étude de l'effet de la limitation de son passage sur la conception et sur les coûts.

3- Présentation de l'algorithme génétique :

3-1- Codage :

Nous avons choisi le codage binaire sans concaténation du chromosome principal, parce que cela nous permet d'éviter la complexité résultant d'un décodage à chaque application des opérateurs génétiques, et lors de calcul d'adaptation. La Figure 3 schématise la disposition des différents gènes sur le chromosome principal et montre la longueur totale du codage de ce dernier, et les longueurs des différents gènes.

3-2- Structure :

Notre algorithme génétique est composé, d'une manière globale (voir Figure 4), de deux générateurs de la population initiale, de quatre opérateurs génétiques (opérateur de sélection, opérateur de croisement, opérateur de mutation et opérateur de pénalisation) et de quatre mécanismes (codage, adaptation, chasse et arrêt). Notre algorithme génétique est composé de trois phases principales : phase d'initiation, d'évolution et d'évaluation.

3-3- Réglage des paramètres génétiques :

- Nombre de gènes : le chromosome principal est composé de neuf (9) gènes.
- Taille du chromosome: la longueur de codage, dans notre cas, est de 113 codes.
- Tailles de populations :
  - Population initiale : nous avons choisi une population initiale de 200 individus pour le générateur aléatoire, et de 720 individus au maximum pour le générateur heuristique.
  - Populations inter-générations : nous avons choisi une population de soixante (60) individus à faire évoluer d'une génération à une autre, jusqu'à la fin. La taille de la population inter-génération doit rester constante jusqu'à la dernière génération.

- Probabilité de croisement P_c: le choix de la probabilité de croisement est décisif, elle doit être élevée (autour de 0,9) afin de bien explorer les sous domaines et favoriser la convergence. Pour effectuer le croisement, nous avons travaillé à deux niveaux distincts:
  - A l'échelle des chromosomes : choisir les gènes sur lesquels on effectue le croisement. Dans notre cas, tous les gènes subissent le croisement.
  - A l'échelle des gènes : choisir la position de point de coupe et la longueur de gène à couper. Pour ceci, la probabilité de croisement sur un gène est de 0,1. La probabilité totale de croisement sur le chromosome devient : 9x0,1=0,9 .
- Nombre de points de coupe sur le chromosome : il doit être supérieur à l'unité. Nous avons choisi le nombre de coupe égal à deux fois le nombre de gènes, c.à.d chaque gène est coupé en deux points.

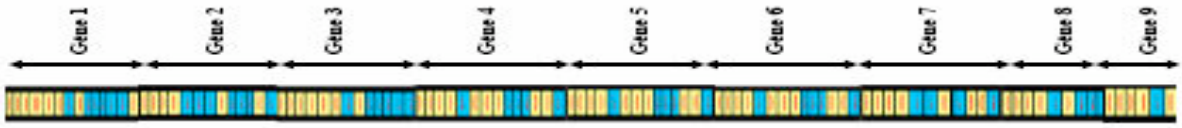


Figure 3 : dispositions des différents gènes sur le chromosome principal.

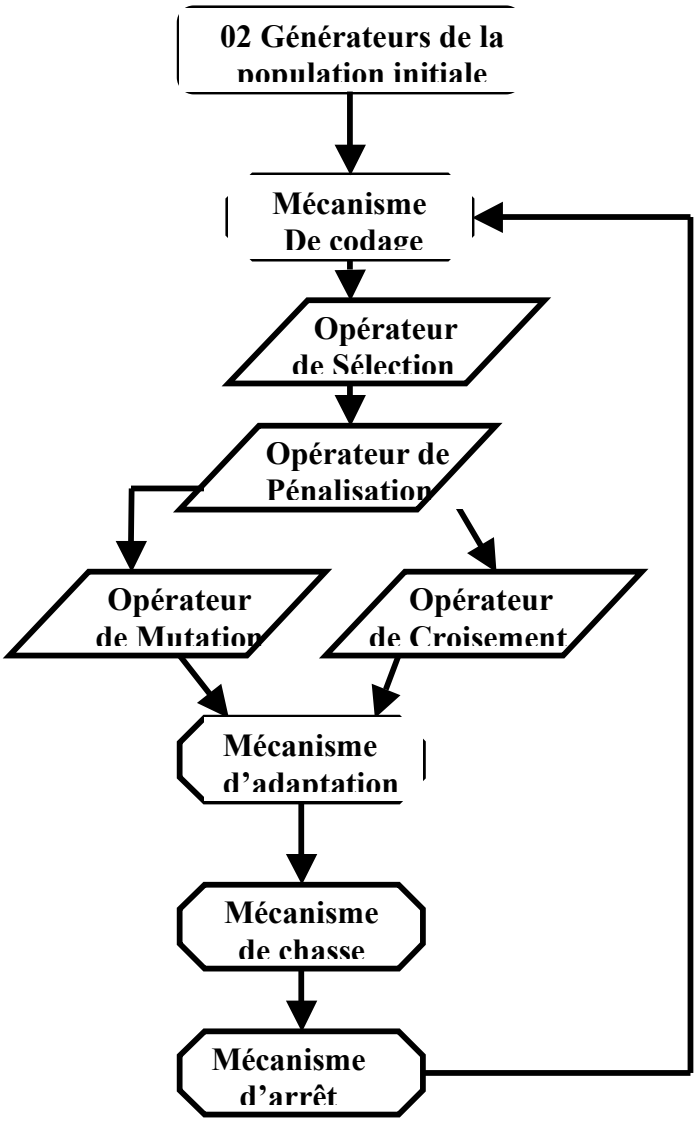


Figure 4 : Différentes composantes et fonctionnement de l'algorithme

❖ Probabilité de mutation P_m : le choix de la probabilité de mutation est aussi important. Elle doit être faible (autour de 0,1) pour bien prospecter l'espace de recherche, éviter des pièges et ne pas modifier complètement le chromosome. Pour effectuer la mutation, nous travaillons à trois niveaux (échelles) : (1) à l'échelle des chromosomes, pour choisir les

gènes qui vont subir une mutation. Dans notre cas, tous les gènes subissent une mutation; (2) à l'échelle des gènes, pour choisir la position de bit à muter. Pour ceci, la probabilité de mutation est de $\frac{1}{10x\text{Nombre de gènes}} = \frac{1}{10.9} = \frac{1}{90} = 0{,}01$

La probabilité totale de mutation sur le chromosome devient : 9x1/90=0,1 ; et (3) à l'échelle des bits, pour choisir le nombre de bits à muter par gène. Nous avons fixé ce nombre à muter à un bit par gène, soit une probabilité de 1/L où L est la longueur (nombre de bits) d'un gène.

- ❖ Critères d'arrêt : le critère d'arrêt est basé sur l'absence d'évolution des dernières générations.
- ❖ Coefficient de pénalité.

4- Résultats et discussion:

4-1- Etude de validation :

Pour l'étude de validation, nous avons testé, étudié et comparé dix-neuf usines de dessalement, se trouvant dans différentes régions du monde, de tailles allant de 2000 à 500 000 m^3/j, avec des salinités de l'eau à dessaler très variées (24 500 à 44 885 ppm), des températures moyennes allant de 18 à 30 °C, des limites imposées sur la salinité totale du produit variable allant de 100 à 500 ppm, des contraintes non linéaires (salinité totale et concentration de Bore du permeat) et des durées de vie très différentes (15 à 40 ans). Les usines sélectionnées sont toutes conçues pour dessaler l'eau de mer par membranes à fibres creuses. La liste des usines, ainsi que leurs capacités, est donnée dans le Tableau 1.

Tableau 1: Illustrations des usines de dessalement étudiées

N°	Usine	Capacité (m^3/j)
1	Tampa,FL,USA [34]	94 000
2	Marbella,Spain [37]	56 400
3	KFNB,AS [37]	7 575
4	Larnaca, Cyprus [34]	40 000
5	Tajura, Libya [38]	38 000
6	Eilat1: Israel [34]	20 000
7	RO-HTR300,China [39]	50 4000
8	- Kuwait [28]	4 546
9	- Israel [40]	38 750
10	Eilat2,Israel [34]	20 000
11	- Singapore 1 [27]	4 500
12	- Singapore 2 [27]	4 500

13	- Argentina 1 [24]	2 000
14	- Argentina 2 [24]	2 000
15	- Argentina 3 [24]	2 000
16	- Argentina 4 [24]	2 000
17	- Argentina 5 [24]	2 000
18	- Argentina 6 [24]	2 000
19	- Argentina 7 [24]	2 000

4-2- Discussions :

Parmi les exemples que nous avons étudiés, il y a six usines existantes en état de fonctionnement, trois en phase d'étude (parmi lesquelles, la plus grande future usine de dessalement à plus de 500 000 m^3/j : la station RO-HTR300 en Chine [39]), et dix études théoriques d'optimisation où les méthodes d'optimisation utilisées antérieurement sont diverses aussi : méthodes déterministes, méthodes à gradient, analyse de scénarios. Le but de notre travail consiste à comparer les résultats de nos calculs d'optimisations avec l'état indiqué de chaque usine, (le prix unitaire de l'eau dessalée surtout, mais aussi la conception (schéma de montage) et paramètres de fonctionnement). Nous avons noté, pour chaque cas, s'il y a lieu, la raison pour l'avantage apporté par l'optimisation (généralement, grâce à soit un meilleur choix de conception (schéma), soit un meilleur choix de paramètres de fonctionnement, ou bien les deux à la fois). L'optimisation conceptuelle par algorithme génétique peut réduire le prix unitaire de l'eau dessalée jusqu'à plus de 60% (voir le Tableau 2 et les Figures 5 et 6). Dans tous les cas, les gains sont supérieurs à 18%. La modélisation conceptuelle (schémas sélectionnés) est aussi bien faite et fondée. De ce fait, les conceptions à deux étages en série rejet, sont évidemment non économiques (cas 13 à 19) : aucun cas de ce type n'a prouvé son utilité. Parmi les dix neuf cas étudiés, l'optimisation a montré que les conceptions de treize usines ont été mal faites (conceptions défaillantes), voir, par exemple, la Figure 5. Pour les six cas restants, l'élévation du prix unitaire de l'eau dessalée, n'est pas due à la conception, mais aux choix non-optimaux des paramètres de fonctionnement.

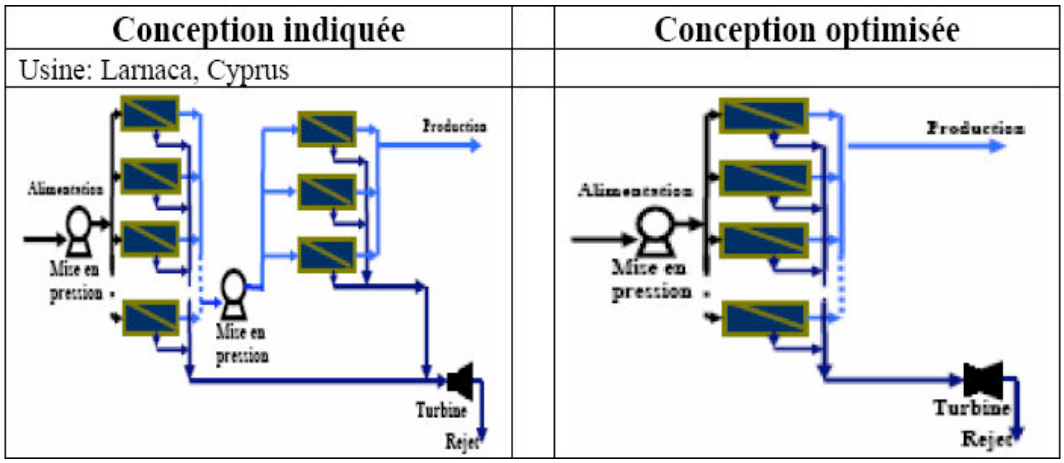


Figure 5: comparaison entre conceptions existante et optimisée par Desaltop (cas de l'usine de Larnaca).

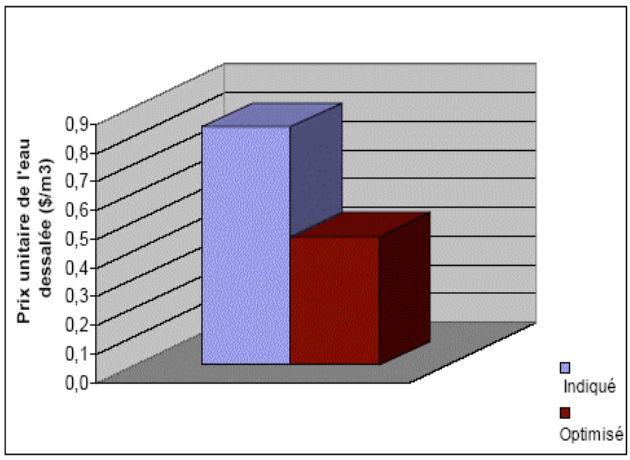


Figure 6: Comparaison entre le prix de l'eau dessalée par une conception existante et celui de la conception optimisée par Desaltop (cas de l'usine de Larnaca)

Tableau 2: synthèse de l'optimisation par DESATOP des différentes usines de dessalement dans plusieurs pays du monde

Usine	Etat de l'usine	Prix indiqué ($/m^3)	Prix Optimisé ($/m^3)	Gain de l'optimisa.	Conception indiquée	Conception optimisée	Conversion indiquée	Conversion optimisée	Pression indiquée KPa	Pression indiquée KPa	Niveau de l'optimis.
Tampa,FL,USA	Existante	0,55	0,3809	30,75%	2 étages	3 étages					CO
Marbella,Spain	Existante	0,4733*	0,3778	20,18%	1	1	0,45	0,4	6850.0	8273.7	CP
KFNB,AS	Existante	0,6433*	0,4836	24,83%	1WER	1	0,35	0,4	6200.0	8273.7	CO-CP
Larnaca, Cyprus	Existante	0,83	0,4474	46,10%	2	1					CO
Tajura, Libya	Existante	0,5793*	0,4593	20,71%	2	3					CO
Eilat1: Israel	Existante	0,72	0,4474	37,86%	1	1	0,5	0,4	-	8273.7	CP
RO-HTR300,China	US	0,42	0,2215	47,26%	1	1	-	0,4	-	8273.7	CP
C.Djannat,Algeria	MG	0,5141	0,5138	0,06%	1	1	0,425	0,426	8274.0	8273.7	CO
- Kuwait	OSA	2,121	1,0609	49,98%	4	3					CO
- Israel	OSA	0,4117*	0,3081	25,16%	1	1	0,5	0,4	6930.0	8273.7	CP
Eilat2,Israel	US	0,81	0,4864	39,95%	1	1	0,45	0,4		8273.7	CP
- Singapore 1	OSA	1,407	1,1322	19,53%	1	1	0,32	0,4	6205.0	8273.7	CP
- Singapore 2	OSA	1,383	1,132	18,15%	2	1					CO
- Argentina 1	DM	0,9095	0,3953	56,54%	2	1					CO
- Argentina 2	DM	0,8695	0,3966	54,39%	2	1					CO
- Argentina 3	DM	0,8313	0,3959	52,38%	2	1					CO
- Argentina 4	DM	0,9095	0,3979	56,25%	2	1					CO
- Argentina 5	DM	0,9536	0,3985	58,21%	2	1					CO
- Argentina 6	DM	1,0108	0,4005	60,38%	2	1					CO
- Argentina 7	DM	1,0028	0,3992	60,19%	2	1					CO

Indications:

* : prix recalculé par DESALTOP
US : usine en état d'étude
MG : étude optimisée par méthode de gradient à pas variable relaxe sur programme excel
OSA : étude optimisée par analyse de scénarios
DM : étude optimisée par modèle déterministe
WER : sans récupération de l'énergie que pour les systèmes à un seul étage.
CO : conception optimisée
CP : paramètres de fonctionnement optimisés.
Les conversions et les pressions ne sont données

5-Conclusion:

Grâce au code d'optimisation multi-critères par algorithme génétique que nous avons élaboré, pour l'optimisation conceptuelle des réseaux d'osmose inverse. Le choix de la méthode a été fait spécialement pour traiter un

problème d'optimisation difficile impliquant une multitude de fonctions objectifs complexes et non-linéaires, avec l'objectif de déterminer les paramètres de fonctionnement optimaux, et d'optimiser le design des réseaux.

L'algorithme génétique adopté est composé essentiellement de deux générateurs de la population initiale (un générateur aléatoire et un autre heuristique), quatre opérateurs génétiques (opérateurs de croisement, de mutation, de sélection et de scaling), et quatre mécanismes (de codage, d'adaptation, de chasse, et d'arrêt). La puissance et l'efficacité de cet algorithme génétique proviennent du choix des paramètres génétiques des différentes composantes ; en effet, une combinaison de deux générateurs est choisie pour améliorer la convergence. Un codage binaire simple sans concaténation a été adopté pour créer les champs de travail des différents opérateurs génétiques, créer la richesse souhaitée et simplifier le travail en évitant la complexité résultant d'un double codage-décodage (Grine, concaténation). Une sélection par la roue biaisée de Goldberg est introduite, pour améliorer les générations au fil d'évolution. Un nombre de 50 à 60 populations à faire évoluer entre générations est jugé suffisant pour la convergence. Le croisement est effectué à une probabilité de 90%, avec des coupes en deux points par gène, pour mieux aider l'opérateur à créer de la diversité. La mutation est effectuée avec une probabilité de 10% sur le chromosome principal, et une probabilité égale à 1/longueur du gène à l'échelle de gène pour rendre la mission de l'exploration de l'espace de recherche plus efficace. La notion du nombre de génération supplémentaire, introduite dans le mécanisme d'arrêt a été efficace pour atteindre un double objectif : améliorer la convergence et réduire le temps de calcul. Un scaling exponentiel (avec un coefficient égal à cinq) a été jugé indispensable pour la convergence, en modifiant le travail de la roulette de Goldberg pour mieux sélectionner les individus semblant nécessaires pour une bonne évolution.

Le code d'optimisation, DESALTOP, a été développé pour optimiser la conception des installations d'osmose inverse, avec le but d'avoir un produit de bonne qualité à un coût moindre. Ce code, qui est composé de trente neuf modules, permet de choisir par optimisation la meilleure conception et les meilleurs paramètres de fonctionnement, en respectant les différentes contraintes techniques et économiques. Il permet aussi de choisir le type de membrane la plus appropriée. En plus de l'optimisation du design, le code Desaltop est capable également de modéliser des installations à conception et paramètres de fonctionnement imposés, de calculer le prix unitaire de l'eau dessalée, et les différents éléments de coût, ce qui fait de lui un outil puissant de décision et de planification des moyens financiers liés au dessalement de l'eau de mer.

Une étude de fiabilité et de validation a été menée, qui a démontré l'efficacité de DESALTOP. Cette étude de fiabilité est effectuée sur 20 installations de dessalement à membrane en fibres creuses existantes dans plusieurs pays.
La puissance du code Desaltop est due à deux éléments principaux :
- la modélisation conceptuelle, montrant que la plupart des usines sont mal conçues.
- et l'algorithme génétique mis en ouvre, qui permet de chercher efficacement la meilleure solution, malgré la non-linéarité et le nombre important de variables et de contraintes.

Références :

1. Masaaki Sekino, Study of an analytical model for hollow fiber reverse osmosis module systems. Desalination, Desalination 100 (1995) 85-97.
2. Victor M.Starov et al., Performance optimization of hollow fiber reverse osmosis membranes, part I. development of theory, J. Membr. Sci. 103 (1995) 257-270.
3. D.Van Gauwbergen et al., Macroscopic fluid flow conditions in spiral-wound membrane elements, Desalination 110 (1997) 287-299.
4. Nader M.Al-Bastaki et al., Predicting the performance of RO membranes, Desalination 132 (2000) 181-187.
5. A.Bergena et al., An experimental assessment of centrifugal membrane separation using spiral wound RO membrane elements, Desalination 154 (2003) 225-232.
6. Abhijit Chatterjee et al., Modeling of a radial flow hollow fiber module and estimation of model parameters using numerical techniques, J. Membr. Sci. 236 (2004) 1-16.
7. M.Metaiche et al., Contribution à la modélisation du facteur de correction de flux de rétention de la membrane «MFRC» de dessalement de l'eau de mer : cas des modules B-10, Desalination 158 (2003) 255-258.
8. M.Metaiche et al., Mathematical modeling of desalination parameters: mono-stage reverse osmosis, Desalination 165 (2004) 153.
9. S.Senthilmurungan et al., Modeling of a spiral-wound module and estimation of model parameters using numerical techniques, Desalination 173 (2005) 269-286.
10. Hoon Hyung et al., A mechanistic study on boron rejection by sea water reverse osmosis membranes, J. Membr. Sci. 286 (2006) 269-278.
11. S.Senthilmurungan et al., Modeling of a radial flow hollow fiber module and estimation of model parameters for aqueous multi-component mixture using numerical techniques, J. Membr. Sci. 279(2006) 466-478.

12. M.Metaiche et al., Contribution à la Modélisation du Prix de Revient de l'eau Dessalée par un Système d'Osmose Inverse Mono étage, Proceeding du Colloque International sur l'Eau, Chlef, 5-6 Février, 2002, pp 270-275.
13. Abdul Sattar Kahdim et al., Modeling of reverse osmosis systems, Desalination 158 (2003) 323-329.
14. Nadder Al-Bastaki et al., Permeate recycle to improve the performance of a spiral-wound RO plant, Desalination 158 (2003) 119-126.
15. M.Metaiche et al., Modelling of Variation with Time of Flow of Membrane Desalination Permeators: B9 and B10 Cases, Procecding du Séminaire International: Eau et Terre, Annaba, 4-6 Décembre, 2004.
16. Abderrahim Abbas, Simulation and analysis of an industrial water desalination plant, Chem. Eng. Process. 44 (2005) 999-1004.
17. M.Metaiche et al., Water desalination price from recent performances: modelling, simulation and analysis, Int. J. Nuclear Desalination Vol. 1, N° 4 (2005), pp.456-465.
18. Haytham I.Qahtani et al., Effect of aging on the performance of RO hollow fiber membranes in a section of an RO plant, Desalination 101 (1995) 177-183.
19. N.G.VOROS et al., Short-cut structural design of reverse osmosis desalination plants, J. Membr. Sci. 127 (1997) 47-68.
20. Nader M.Al-Bastaki et al., Modeling an industrial reverse osmosis unit, Desalination 126(1999)33-39.
21. Masahide Tanaguchi et al., Behavior of a reverse osmosis plant adopting a brine conversion two-stage process and its computer simulation, J. Membr. Sci. 183 (2001) 249-257.
22. A.Villafafila and al., Fresh water by reverse osmosis based desalination: simulation and optimisation, Desalination 155 (2003) 1-13.
23. IDA, The 2004 IDA Worldwide Desalination Plants Inventory Report n°18, published by Wangnick Consulting, 2004.
24. Marian G. Marcovecchio et al., optimal design of reverse osmosis networks for seawater desalination: modeling and algorithm, Desalination 184 (2005) 259–271.
25. Chandan Guria et al., Multi-objective optimization of reverse osmosis desalination units using different adaptations of the non-dominated sorting genetic algorithm (NSGA), Computers and Chemical Engineering 29 (2005) 1977–1995.
26. A.Malek et al., A lumped transport parameter approach in predicting B10 RO permeator performance, Desalination, 99 (1994) 19-38.
27. A.Malek et al., Design and Economics of Seawater Desalination, Desalination 105(1996).
28. A.A.J.Al Zubaidi, Parametric Cost Analysis Study of Seawater Reverse Osmosis Systems Design in Kuwait, Desalination, 76(1989).

29. Fazilet, Optimization of reverse osmosis membrane networks, PhD Thesis, University of New South Wales, Australia, 2000.
30. I.S.Al-Mutaz et al., Optimum Design for a Hybrid Desalination Plant, Desalination 76 (1989).
31. A.M. Helal et al., Optimal design of hybrid RO/MSF desalination plants. Part III. Snsitivity analysis, Desalination, 169(2004) 43–60.
32. Abderrahim Abbas et al., Modeling of an RO water desalination unit using neural networks, Chem.Eng.J. 114(2005)139-143.
33. Andreas Poullikkas, Optimization algorithm for reverse osmosis desalination economics, Desalination 133 (2001) 75-81.
34. Mark Wilf et Kenneth Klinko, Optimisation of seawater RO systems design, Desalination 138(2001) 299-306.
35. Jim Smart et al., Performance optimization of hollow fiber reverse osmosis membranes. Part II. Comparative study of flow configurations, J.Membr.Sci. 119(1996) 117-128.
36. H.Ohya et al., An analysis of reverse osmotic characteristics of B-9 hollow fiber module, Desalination 21 (1977) 257-274).
37. Vashek Polasek et Al., Conversion from hollow fiber to spiral technology in large seawater RO systems- Process design and economics, Desalination 156(2003) 239-247.
38. Ibrahim M.El-Azizi et al., Design criteria of 10,000 m^3/d SWRO desalination plant of Tajura, Libya, Desalination 153(2002) 273-279.
39. Li Tian et Al., Economic pre-feasibility study of seawater desalination for a high-temperature gas-cooled reactor by reverse osmosis, Desalination 157(2003) 253-258.
40. P.Glueckstern et al., Advanced concept of large seawater desalination systems for Israel, Desalination 119(1998) 33-45.

Méthode duale de support pour la résolution des problèmes quadratiques convexes à variables bornées

B. BRAHMI et M. O. BIBI

Laboratoire LAMOS, Université A. MIRA, 06000 Béjaia, Algérie.
bra_belka@yahoo.fr mohandbibi@yahoo.fr

Abstract. Dans ce papier, on présentera une méthode duale pour la résolution des problèmes quadratiques convexes à variables bornées. Cette méthode présente deux phases, où la première consiste à construire un support initial pour le problème, appelé "support coordinateur" ; la seconde est la méthode proprement dite qui permet le passage d'une itération à une autre. Elle a l'avantage de traiter les contraintes de bornes telles qu'elles se présentent et traite aussi les problèmes quadratiques d'une manière générale, et ce, contrairement à d'autres méthodes qui sont uniquement développées pour le cas d'un programme quadratique strictement convexe.

Mots clés : programmation quadratique convexe, forme quadratique semi-définie positive, dualité en programmation quadratique convexe, support coordinateur, estimation de suboptimalité, méthode duale de support.

1 Introduction

La programmation quadratique est une théorie autonome de la programmation mathématique, qui consiste à optimiser une fonction quadratique sous des contraintes linéaires. Cette branche est très importante du point de vue application et du point de vue théorique. En effet, les modèles quadratiques sont souvent rencontrés dans plusieurs domaines, comme par exemple la sélection d'un portefeuille, la régression linéaire, machine à support de vecteur, ou les problèmes de contrôle optimal, $\cdots$ etc.

Au point de vue théorique, elle est une transition naturelle entre la programmation linéaire et non linéaire. Les algorithmes développés pour le cas de l'optimisation non linéaire reposent essentiellement sur l'approche quadratique, qui utilise les méthodes de programmation quadratique séquentielle (PQS) [11].

Dans la littérature, plusieurs approches ont été proposées pour la résolution des problèmes quadratiques convexes [9-14]. Dans cet article, nous développons une autre approche, basée les méthodes adaptées de support [1-3].

2 Position du problème et définitions

Considérons le problème de programmation quadratique convexe suivant :

$$\begin{cases} F(x) = \frac{1}{2}x'Dx + c'x \longrightarrow \min, \\ Ax = b, \\ d^- \leq x \leq d^+, \end{cases} \tag{1}$$

où c, d^-, d^+ et x sont des n-vecteurs ; b est un m-vecteur ; la matrice A est d'ordre $m \times n$, avec rang$(A) = m < n$. La matrice D est carrée d'ordre n, supposée symétrique et semi-définie positive ($D \geq 0$). Le symbole (') est celui de la transposition.
Les ensembles d'indices des lignes et colonnes de A sont respectivement notés par :
$I = \{1, 2, \cdots, m\}$, $J = \{1, 2, \cdots, n\} = J_B \cup J_N$, avec $J_B \cap J_N = \emptyset$, $|J_B| = m$.

Définition 1.
- Un vecteur x vérifiant les contraintes du problème (1) est appelé *plan* ou *solution réalisable* du problème.
- Un plan x^0 est dit *optimal* si

$$F(x^0) = \frac{1}{2} x^{0\prime} D x^0 + c' x^0 = \min_x F(x),$$

où x est pris parmi tous les plans du problème.
- Pour un nombre $\epsilon > 0$ donné, un plan x^ϵ est dit ϵ-optimal ou suboptimal si

$$F(x^\epsilon) - F(x^0) \leq \epsilon,$$

où x^0 est une solution optimale du problème (1).
- L'ensemble $J_B \subset J$ tel que $|J_B| = m$ est appelé support des contraintes du problème (1) si

$$\det A_B = \det A(I, J_B) \neq 0.$$

- Le couple $\{x, J_B\}$ formé du plan x et du support J_B est appelé plan de support des contraintes. Il est dit non dégénéré si

$$d_j^- < x_j < d_j^+, \forall j \in J_B.$$

Définissons le vecteur des potentiels u ainsi que le vecteur des estimations E de la manière suivante :

$$u' = g'_B A_B^{-1}, \quad E' = E'(J) = g' - u' A = (E'_B, E'_N),$$

où

$$E'_N = g'_N(x) - u' A_N, \quad E'_B = 0,$$

avec $g = g(J) = g(x) = Dx + c$ représente le gradient de la fonction F au point x, avec $g' = (g'_B, g'_N)$.
Le critère d'optimalité du problème (1) est donné dans le théorème suivant :

Théorème 1. *[1,3] Soit $\{x, J_B\}$ un plan de support des contraintes du problème (1). Alors les relations :*

$$\begin{cases} E_j \geq 0, \textit{ pour } & x_j = d_j^-, \\ E_j \leq 0, \textit{ pour } & x_j = d_j^+, \\ E_j = 0, \textit{ pour } & d_j^- < x_j < d_j^+, \; j \in J_N = J_S \cup J_{NN}, \end{cases} \tag{2}$$

sont suffisantes pour l'optimalité du plan x. Ces mêmes relations sont aussi nécessaires, si $\{x, J_B\}$ est non dégénéré.

Définition 2. – L'ensemble $J_S \subset J_N = J \backslash J_B$ est appelé support de la fonction objectif F du problème (1) si

$$\det M_S = \det M(J_S, J_S) \neq 0,$$

où

$$M = M(J_N, J_N) = Z'DZ \quad \text{et} \quad Z = Z(J, J_N) = \begin{pmatrix} -A_B^{-1} A_N \\ I_N \end{pmatrix}, \quad (3)$$

I_N étant la matrice unité d'ordre $(n-m)$.

– L'ensemble $J_P = \{J_B, J_S\}$ formé du support des contraintes J_B et du support de la fonction objectif F est appelé *support du problème (1).*

3 Problème dual et propriétés

D'après la théorie de la dualité en programmation quadratique convexe [4], le problème dual associé au problème primal (1) est le programme quadratique concave suivant :

$$\begin{cases} \varphi(\lambda) = -\frac{1}{2}\kappa' D\kappa + b'y + v'd^- - w'd^+ \longrightarrow \max, \\ D\kappa + c - A'y - v + w = 0, \\ \lambda = (\kappa, y, v, w),\ \kappa \in \mathbb{R}^n, y \in \mathbb{R}^m,\ v \geq 0,\ w \geq 0. \end{cases} \quad (4)$$

Ce programme est intéressant à résoudre puisque les variables κ et y sont sans restriction de signe. Introduisons alors les définitions suivantes pour le problème dual :

Définition 3. – Le quadruplet $\lambda = (\kappa, y, v, w)$ vérifiant toutes les contraintes du problème (4) est appelé *plan dual* de ce problème. Le n-vecteur $\delta = D\kappa + c - A'y$ est appelé vecteur *coplan*, associé au plan dual λ.

– Un plan dual $\lambda^0 = (\kappa^0, y^0, v^0, w^0)$ est dit *optimal* si

$$\varphi(\lambda^0) = -\frac{1}{2}\kappa^{0\prime} D\kappa^0 + b'y^0 + v^{0\prime}d^- - w^{0\prime}d^+ = \max_{\lambda} \varphi(\lambda),$$

où $\lambda = (\kappa, y, v, w)$ est pris parmi tous les plans duaux du problème (4). Le n-vecteur correspondant $\delta^0 = D\kappa^0 + c - A'y^0$ est alors dit coplan optimal.

– Le couple $\{\lambda, J_P\}$ formé du plan dual λ et du support du problème J_P est appelé plan dual de support du problème (4). De même, la paire $\{\delta, J_P\}$ est appelée coplan de support pour le problème (4).

3.1 Vecteur pseudo-plan et support coordinateur

Soit J^+ et J^- deux sous-ensembles d'indices de J_{NN}, tels que :

$$J^+ \cup J^- = J_{NN}, \quad J^+ \cap J^- = \emptyset, \quad J_{NN} = J_N \backslash J_S.$$

Construisons le n-vecteur $\kappa = \kappa(J) = (\kappa(J_B), \kappa(J_S), \kappa(J^+), \kappa(J^-))$ tel que :

$$\begin{cases} \kappa(J^+) = d^-(J^+), \\ \kappa(J^-) = d^+(J^-), \end{cases} \quad (5)$$

et où $\kappa(J_B)$ et $\kappa(J_S)$ se calculent à partir des deux relations suivantes :

$$A\kappa = b \quad \text{et} \quad \delta(J_P) = 0.$$

La relation $A\kappa = b$ s'écrit aussi sous la forme :

$$A_B\kappa_B + A_S\kappa_S + A_{NN}\kappa_{NN} = b \Longrightarrow \kappa(J_B) = A_B^{-1}(b - A_S\kappa_S - A_{NN}\kappa_{NN}). \quad (6)$$

Le vecteur κ vérifiant le système $A\kappa = b$ peut être écrit sous la forme suivante :

$$\kappa = Yb + Z\kappa_N, \quad (7)$$

où $\kappa_N = \kappa(J_S \cup J_{NN})$; $Z = Z(J, J_N)$ est une matrice de dimension $n \times (n - m)$, définie par la relation (3) et qui vérifie aussi $AZ = 0$; la matrice $Y = Y(J, I) = \begin{pmatrix} A_B^{-1} \\ 0 \end{pmatrix}$ est de dimension $n \times m$. En multipliant le vecteur coplan par la matrice Z' et en utilisant l'expression (7), on aura :

$$\begin{aligned} Z'\delta &= Z'D(Yb + Z\kappa_N) + Z'c - Z'A'y \\ &= Z'DYb + Z'DZ\kappa_N + Z'c \\ &= Z'D(J, J_B)A_B^{-1}b + M\kappa_N + Z'c. \end{aligned}$$

Pour les ensembles d'indices $J_S \subset J_N$ et $J_P \subset J$, on aura :

$$\begin{aligned} Z'(J_S, J_P)\delta(J_P) &= Z'(J_S, J_P)D(J_P, J_B)A_B^{-1}b + M(J_S, J_N)\kappa_N + Z'(J_S, J_P)c(J_P) \\ &= Z'(J_S, J_P)\left[D(J_P, J_B)A_B^{-1}b + c(J_P)\right] + M(J_S, J_S)\kappa(J_S) + \\ &\quad M(J_S, J_{NN})\kappa(J_{NN}); \end{aligned}$$

d'où

$$\kappa_S = -M_S^{-1}Z'(J_S, J_P)\left[D(J_P, J_B)A_B^{-1}b + c(J_P) - \delta(J_P)\right] - M_S^{-1}M(J_S, J_{NN})\kappa_{NN}.$$

Puisque $\delta(J_P) = 0$, alors il s'ensuit que

$$\kappa_S = -M_S^{-1}Z'(J_S, J_P)\left[D(J_P, J_B)A_B^{-1}b + c(J_P)\right] - M_S^{-1}M(J_S, J_{NN})\kappa_{NN}. \quad (8)$$

Le vecteur κ défini par les formules (5), (6) et (8) est alors appelé *pseudo-plan*, associé au support $J_P = \{J_B, J_S\}$. De plus, s'il vérifie les inégalités

$$d_j^- \leq \kappa_j \leq d_j^+, \text{ pour } j \in J_P = J_B \cup J_S,$$

alors ce dernier est un plan pour le problème primal (1).

Définition 4 (support coordinateur). L'ensemble $J_C = \{J_P, J^+, J^-\}$ est appelé support coordinateur s'il existe un pseudo-plan κ tel que

$$\begin{cases} \delta_j \geq 0, \text{ pour} & j \in J^+, \\ \delta_j < 0, \text{ pour} & j \in J^-. \end{cases} \quad (9)$$

On dit dans ce cas que le pseudo-plan κ est associé au support coordinateur J_C.

Définition 5 (plan dual et coplan de support coordinateur). Soit λ un plan dual du problème (4) et δ son coplan correspondant, accompagnant un support coordinateur J_C. Les paires $\{\lambda, J_C\}$ et $\{\delta, J_C\}$ sont alors appelées respectivement plan dual et coplan de support coordinateur du problème (4).

Définition 6 (plan dual de support coordinateur non dégénéré). Le plan dual de support coordinateur $\{\lambda, J_C\}$ est dit non dégénéré si

$$\delta_j \neq 0, \quad j \in J_{NN}.$$

Parmi les différents plans duaux du problème (4), on ne s'intéresse, dans ce travail, qu'aux plans duaux accordés $\lambda = (\kappa, y, v, w)$ pour lesquels les n-vecteurs v et w sont construits comme suit :

$$\begin{cases} v_j = \delta_j, \, w_j = 0, & \text{si} \quad \delta_j \geq 0, \\ v_j = 0, \;\; w_j = -\delta_j, & \text{si} \quad \delta_j < 0, \quad j \in J. \end{cases} \tag{10}$$

Un plan dual accordé possède des propriétés très intéressantes, qui sont données dans les lemmes suivants :

Lemme 1. *[8] Pour tout plan dual quelconque $\lambda = (\kappa, y, v, w)$, il existe un plan dual accordé $\bar{\lambda} = (\bar{\kappa}, \bar{y}, \bar{v}, \bar{w})$, tel que*

$$\varphi(\lambda) \leq \varphi(\bar{\lambda}). \tag{11}$$

Lemme 2. *[8] Pour un plan dual accordé quelconque $\lambda = (\kappa, y, v, w)$, et un plan primal x arbitraire, on a :*

$$\varphi(\lambda) \leq F(x).$$

Lemme 3. *[8] Si un plan dual accordé $\lambda = (\kappa, y, v, w)$ est associé à un support coordinateur J_C, on a alors la relation :*

$$\varphi(\lambda) = F(\kappa). \tag{12}$$

3.2 Construction d'un support coordinateur initial

La construction d'un support coordinateur initial J_C est nécessaire pour la formulation des critères d'optimalité et de suboptimalité, ainsi que pour la phase I de la méthode duale de support. Avant de décrire une procédure pour sa construction, considérons le théorème suivant qui permet de prouver son existence.

Théorème 2. *[8] Le problème dual (4) admet des solutions optimales. Parmi ces dernières, il existe au moins un plan dual accordé optimal, associé à un support coordinateur J_C^0.*

La procédure ci-après permet de construire un support coordinateur $J_C = \{J_B, J_S, J^+, J^-\}$, ainsi que son pseudo-plan et son coplan correspondants. Ces étapes sont les suivantes :

(1) Construction d'un support $J_P = \{J_B, J_S\}$ pour le problème primal (1) et son pseudo-plan correspondant :

– Choisir $J_B \subset J$, $|J_B| = m$ tel que : det $(A_B) \neq 0$.
– Calculer les matrices suivantes :

$$Z = Z(J, J_N) = \begin{pmatrix} -A_B^{-1} A_N \\ I_N \end{pmatrix}, \quad M = M(J_N, J_N) = Z' D Z, \tag{13}$$

– Sélectionner $J_S \subset J_N = J \backslash J_N$ tel que : det $M_S =$ det $M(J_S, J_S) \neq 0$.
– Construire le pseudo-plan correspondant $\kappa = (\kappa_B, \kappa_S, \kappa_{NN})'$:

$$\begin{cases} d_j^- \leq \kappa_j \leq d_j^+, \; j \in J_{NN}, \\ \kappa_S = -M_S^{-1} Z'(J_S, J_P)[D(J_P, J_B) A_B^{-1} b + c(J_P)] - M_S^{-1} M(J_S, J_{NN}) \kappa_{NN}, \\ \kappa_B = A_B^{-1}(b - A_N \kappa_N). \end{cases}$$

– Calculer le vecteur coplan $\delta = (\delta_P, \delta_{NN})'$:

$$\delta_P = 0, \quad \delta_{NN} = D(J_{NN}, J)\kappa + c_{NN} - A'_{NN} y, \text{ avec } y = [A'_B]^{-1}(D(J_B, J)\kappa + c_B).$$

– Poser

$$J^+ = \{j \in J_{NN}, \, \delta_j \geq 0\}, \quad J^- = \{j \in J_{NN}, \, \delta_j < 0\}.$$

(2) Application des tests de coordination

- **Si** $\kappa_j = d_j^-$, $j \in J^+$ **et** $\kappa_j = d_j^+$, $j \in J^-$ **alors**
  $\{\delta, J_C\}$ est un coplan de support coordinateur. Arrêter le procédé.
- **Sinon**
  – Construire un nouveau pseudo-plan $\bar{\kappa}$ et son coplan $\bar{\delta}$ correspondant :

$$\bar{\kappa} = \kappa + \sigma^0 l, \quad \bar{\delta} = \delta + \sigma^0 t, \text{ avec } t = Dl - A'q, \; \bar{y} = y + \sigma^0 q,$$

  où les n-vecteurs l et t représentant respectivement les directions d'amélioration du pseudo-plan et de son coplan sont déterminés comme suit :

$$\begin{cases} l_j = d_j^- - \kappa_j, \quad \text{si } j \in J^+, \\ l_j = d_j^+ - \kappa_j, \quad \text{si } j \in J^-, \\ l_S = -M_S^{-1} M(J_S, J_{NN}) l_{NN}, \\ l_B = -A_B^{-1} A_N l_N, \end{cases}, \quad \begin{cases} t_B = 0, \, t_S = 0, \\ t_{NN} = D(J_{NN}, J) l - A'_{NN} q, \\ q = [A_B^{-1}]' D(J_B, J) l. \end{cases}$$

  – Calculer le pas optimal σ^0 :

$$\sigma^0 = \min\{1, \sigma_{j_0}\},$$

  où σ_{j_0} est déterminé de telle manière que δ_{NN} et $\bar{\delta}_{NN}$ ne changent pas de signe :

$$\sigma_{j_0} = \min_{j \in J_{NN}} \sigma_j, \text{ avec } \sigma_j = \begin{cases} -\dfrac{\delta_j}{t_j}, & \text{si } \delta_j t_j < 0, \\ +\infty, & \text{si } \delta_j t_j \geq 0, \; j \in J_{NN}. \end{cases} \tag{14}$$

- **Si** $\sigma^0 = 1$ **alors**
  $J_C = \{J_B, J_S, J^+, J^-\}$ est un support coordinateur initial et $\bar{\kappa} = \kappa + l$ son pseudo-plan associé.

- **Sinon** ($\sigma^0 = \sigma_{j_0} < 1, j_0 \in J_{NN}$), poser :

$$\begin{array}{l} \bar{J}_S = J_S \cup j_0, \quad \bar{J}_B = J_B, \quad \bar{J}_P = \bar{J}_B \cup \bar{J}_S, \\ \bar{J}^+ = J^+ \backslash j_0, \quad \bar{J}^- = J^-, \text{ si } j_0 \in J^+, \\ \bar{J}^+ = J^+, \quad \bar{J}^- = J^- \backslash j_0, \text{ si } j_0 \in J^-. \end{array}$$

Aller à (2).

Remarque 1.
- Si $J_{NN} = \emptyset$ alors $J_C = \{J_P = J_B \cup J_S, J^+ = \emptyset, J^- = \emptyset\}$ est un support coordinateur pour le problème (1) et par conséquent, il n'est pas nécessaire d'exécuter l'étape (2) de l'algorithme ci-dessus.
- La procédure proposée est finie, puisque elle permet de construire au pire des cas un support coordinateur en $(n - m)$ itérations, lorsque on démarre avec $J_S = \emptyset$.

3.3 Critère d'optimalité

Puisque $\delta(J_P) = 0$, par construction, alors le critère d'optimalité pour le problème dual (4) est le suivant :

Théorème 3. *[8] Les relations*

$$d_j^- \leq \kappa_j \leq d_j^+, \quad \forall j \in J_P = J_B \cup J_S, \tag{15}$$

sont suffisantes pour l'optimalité du coplan de support coordinateur $\{\delta, J_C\}$. Elles sont aussi nécessaires dans le cas où $\{\delta, J_C\}$ est non dégénéré. Le pseudo-plan κ correspondant au coplan optimal est alors une solution optimale du problème primal (1).

4 Algorithme de la méthode

Le principe de la méthode consiste à transformer le couple $\{\delta, J_C\} \longrightarrow \{\bar{\delta}, \bar{J}_C\}$ et cela en deux étapes ; la première est la construction d'un nouveau coplan $\bar{\delta}$ telle que $\varphi(\bar{\lambda}) \geq \varphi(\lambda)$, où λ et $\bar{\lambda}$ représentent respectivement les plans duaux associés aux coplans δ et $\bar{\delta}$. La deuxième étape consiste à transformer $J_C \longrightarrow \bar{J}_C$ de telle manière que $\bar{J}_C$ soit un support coordinateur.

L'algorithme de la méthode présente les étapes suivantes :

(0) Construire un coplan de support coordinateur initial $\{\delta, J_C\}$ en utilisant la phase I de la méthode .

(1) Tester l'optimalité du coplan de support coordinateur $\{\delta, J_C\}$.
- **Si** les relations d'optimalité (15) sont vérifiées, alors κ est optimal pour le problème primal (1). Arrêter l'algorithme.
- **Sinon** pour les indices non optimaux $J_P^{\text{no}} \subset J_P$, où J_P est le support du problème, calculer les nombres suivants :

$$\alpha_j = \begin{cases} \kappa_j - d_j^-, \text{ si } \kappa_j < d_j^-, \\ \kappa_j - d_j^+, \text{ si } \kappa_j > d_j^+, j \in J_P^{no}. \end{cases}$$

Choisir un indice $j_1 \in J_P^{no}$ vérifiant :

$$\alpha_{j_1} = \max\{|\alpha_j|,\ j \in J_P^{no}\}.$$

Aller à **(2)**.

(2) Construction d'un nouveau coplan et de son pseudo-plan associé :

$$\bar{\delta} = \delta + \sigma^0 t,\ \bar{\kappa} = \kappa + \sigma^0 l,\ \bar{y} = y + \sigma^0 q,\ \sigma^0 \geq 0.$$

– Calculer les directions d'amélioration $l = (l_B, l_S, l_{NN})'$ et $t = (t(J_P), t(J_{NN}))'$ de la manière suivante :

$$\begin{cases} l_{NN} = 0, \\ l_S = -M_S^{-1} Z'(J_S, j_1)\text{sign}\alpha_{j_1}, \\ l_B = -A_B^{-1} A_S l_S, \end{cases} \quad \begin{cases} t_{j_1} = -\text{sign}\alpha_{j_1},\ t(J_P \backslash j_1) = 0, \\ t_{NN} = D(J_{NN}, J_P) l_P - A'_{NN} q; \\ q = [A_B^{-1}]'[D(J_B, J_P) l_P - t_B]. \end{cases}$$

– Calculer le pas optimal σ^0 :

$$\sigma^0 = \min\{\sigma_{j_1}, \sigma_{j_0}\}, \tag{16}$$

où σ_{j_0} représente la valeur maximale du pas pour lesquels δ_{NN} et $\bar{\delta}_{NN}$ gardent le même signe.

$$\sigma_{j_0} = \min_{j \in J_{NN}} \sigma_j,\ \text{avec } \sigma_j = \begin{cases} -\dfrac{\delta_j}{t_j},\ \text{si } \delta_j t_j < 0, \\ \infty, \quad \text{si } \delta_j t_j \geq 0,\ j \in J_{NN}, \end{cases}$$

et σ_{j_1} est le pas maximal pour lequel l'indice j_1 devient optimal.

$$\sigma_{j_1} = \begin{cases} -\dfrac{\alpha_{j_1}}{l_{j_1}},\ \text{si } l_{j_1} \neq 0, \\ \infty, \quad \text{si } l_{j_1} = 0. \end{cases}$$

- **Si** $\sigma^0 = \infty$ **alors** le problème dual (4) est non borné et par conséquent son primal (1) est irréalisable. Arrêter l'algorithme.
- **Sinon** ($\sigma^0 < \infty$), calculer

$$\bar{\delta} = \delta + \sigma^0 t,\ \bar{\kappa} = \kappa + \sigma^0 l,\ \bar{y} = y + \sigma^0 q.$$

Aller à **(3)**.

(3) Construction d'un nouveau support coordinateur $\bar{J}_C = \{\bar{J}_B, \bar{J}_S, \bar{J}^+, \bar{J}^-\}$

a) Si $\sigma^0 = \sigma_{j_1}, j_1 \in J_P = J_B \cup J_S$ **alors** l'indice j_1 est optimal et on pose :

$$\begin{cases} \bar{J}^+ = J^+ \cup j_1,\ \bar{J}^- = J^-, & \text{si } l_{j_1} > 0, \\ \bar{J}^+ = J^+, \quad \bar{J}^- = J^- \cup j_1, & \text{si } l_{j_1} < 0. \end{cases}$$

- **Si** $j_1 \in J_S$ alors

$$\bar{J}_B = J_B, \quad \bar{J}_S = J_S \backslash j_1.$$

- **Sinon** ($j_1 \in J_B$). Choisir un indice $j_* \in J_S$ tel que $z_{j_1 j_*} = Z(j_1, j_*) \neq 0$ et poser :

$$\bar{J}_B = (J_B \backslash j_1) \cup j_*, \quad \bar{J}_S = J_S \backslash j_*.$$

Aller à **(1)** en partant de $\{\bar{\delta}, \bar{J}_C\}, \bar{\kappa}$.

b) Sinon ($\sigma^0 = \sigma_{j_0}$), l'indice j_1 n'est pas optimal. Calculer le réel suivant :

$$\bar{\alpha}_{j_1} = \bar{\kappa}_{j_1} - d^-_{j_1} < 0 \text{ si } t_{j_1} = 1, \quad \bar{\alpha}_{j_1} = \bar{\kappa}_{j_1} - d^+_{j_1} > 0 \text{ si } t_{j_1} = -1.$$

i) Si $\eta_0 = M(j_0, j_0) - M(j_0, J_S) M_S^{-1} M(J_S, j_0) \neq 0$ **alors**

$$\bar{J}_S = J_S \cup j_0, \quad \bar{J}_B = J_B, \quad \bar{J}_{NN} = J_{NN} \backslash j_0,$$

$$\bar{J}^+ = J^+ \cap \bar{J}_{NN}, \; \bar{J}^- = J^- \cap \bar{J}_{NN}.$$

Aller à **(2)**, en partant de $\{\bar{\delta}, \bar{J}_C\}, \bar{\kappa}, j_1, \bar{\alpha}_{j_1}$.

ii) Sinon ($\eta_0 = M(j_0, j_0) - M(j_0, J_S) M_S^{-1} M(J_S, j_0) = 0$)

- Construire un nouveau pseudo-plan $\bar{\bar{\kappa}} = \bar{\kappa} + \theta^0 \tilde{l}$:
  - Calculer la direction corrigée $\tilde{l}$:

$$\begin{cases} \tilde{l}(J_{NN} \backslash j_0) = 0, \\ \tilde{l}_{j_0} = 1 \text{ si } j_0 \in J^+, \quad \tilde{l}_{j_0} = -1 \text{ si } j_0 \in J^-, \\ \tilde{l}(J_S) = -M_S^{-1} M(J_S, j_0) \tilde{l}_{j_0}, \\ \tilde{l}(J_B) = -A_B^{-1} A(I, J_S \cup j_0) \tilde{l}(J_S \cup j_0). \end{cases}$$

  - Calculer le nouveau pas θ^0 :

$$\theta^0 = \min\{\theta_{j_0}, \theta_{j_1}\}, \quad \text{avec} \quad \theta_{j_0} = d^+_{j_0} - d^-_{j_0}, \quad \theta_{j_1} = -\frac{\bar{\alpha}_{j_1}}{\tilde{l}_{j_1}}.$$

  - Poser

$$\bar{\bar{\kappa}} = \bar{\kappa} + \theta^0 \tilde{l}, \quad \bar{\bar{\delta}} = \delta(\bar{\bar{\kappa}}) = \bar{\delta}.$$

✓ **Si** $\theta^0 = \theta_{j_1}$ **alors**

- **Si** $j_1 \in J_S$ **alors**

$$\bar{J}_B = J_B, \quad \bar{J}_S = (J_S \backslash j_1) \cup j_0, \quad \bar{J}_{NN} = J_N \backslash \bar{J}_S,$$

$$\begin{cases} \bar{J}^+ = J^+ \cap \bar{J}_{NN} \cup j_1, \; \bar{J}^- = J^- \cap \bar{J}_{NN}, & \text{si } t_{j_1} = 1, \\ \bar{J}^+ = J^+ \cap \bar{J}_{NN}, & \bar{J}^- = J^- \cap \bar{J}_{NN} \cup j_1, \text{ si } t_{j_1} = -1. \end{cases}$$

- **Sinon** ($j_1 \in J_B$). On pose :

$$\bar{J}_B = (J_B \backslash j_1) \cup j_0, \quad \bar{J}_S = J_S, \quad \bar{J}_N = J \backslash \bar{J}_B, \quad \bar{J}_{NN} = \bar{J}_N \backslash \bar{J}_S,$$

$$\begin{cases} \bar{J}^+ = J^+ \cap \bar{J}_{NN} \cup j_1, \; \bar{J}^- = J^- \cap \bar{J}_{NN}, & \text{si } t_{j_1} = 1, \\ \bar{J}^+ = J^+ \cap \bar{J}_{NN}, & \bar{J}^- = J^- \cap \bar{J}_{NN} \cup j_1, \text{ si } t_{j_1} = -1. \end{cases}$$

Aller à **(1)**, en partant de $\{\bar{\bar{\delta}}, \bar{J}_C\}, \bar{\bar{\kappa}}$.

✓ **Sinon** ($\theta^0 = \theta_{j_0}$). Calculer le nombre :

$$\bar{\bar{\alpha}}_{j_1} = \bar{\bar{\kappa}}_{j_1} - d^-_{j_1} < 0 \text{ si } t_{j_1} = 1, \quad \bar{\bar{\alpha}}_{j_1} = \bar{\bar{\kappa}}_{j_1} - d^+_{j_1} > 0 \text{ si } t_{j_1} = -1,$$

et poser :

$$\bar{J}_P = J_P, \quad \begin{cases} \bar{J}^+ = J^+ \backslash j_0, & \bar{J}^- = J^- \cup j_0, \text{ si } t_{j_1} = 1, \\ \bar{J}^+ = J^+ \cup j_0, & \bar{J}^- = J^- \backslash j_0, \text{ si } t_{j_1} = -1. \end{cases}$$

Aller à **(2)**, en partant de $\{\bar{\bar{\delta}}, \bar{J}_C\}, \bar{\bar{\kappa}}, j_1, \bar{\bar{\alpha}}_{j_1}$.

(4) Fin de l'algorithme.

4.1 Exemple numérique

Considérons le problème quadratique suivant :

$$\begin{cases} F(x) = 4x_1^2 - 4x_1x_2 + 2x_2^2 + 2x_1 + x_2 - 3x_3 - x_4 \longrightarrow \min, \\ x_1 \ -4x_2 + x_3 \qquad = 3, \\ 2x_1 + x_2 \qquad + x_4 = 4, \\ -2 \le x_1 \le 2,\ 0 \le x_2 \le 4,\ 2 \le x_3 \le 5,\ -3 \le x_4 \le 6. \end{cases} \tag{17}$$

Soit $J_B = \{3, 4\}$ et $J_N = J \backslash J_B = \{1, 2\}$; $A_B = (a_3, a_4) = I_2$, $A_N = \begin{pmatrix} 1 & -4 \\ 2 & 1 \end{pmatrix}$.
Calcul des matrices suivantes :

$$Z = \begin{pmatrix} -A_B^{-1}A_N \\ I_N \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 4 \\ -2 & -1 \end{pmatrix},$$

$$M = Z'DZ = \begin{pmatrix} 8 & -4 \\ -4 & 4 \end{pmatrix} > 0,\ M_S^{-1} = \begin{pmatrix} 1/4 & 1/4 \\ 1/4 & 1/2 \end{pmatrix}.$$

Comme $M > 0$ alors nous prenons $J_S = J_N = \{1, 2\}$, $J_{NN} = J^+ \cup J^- = \emptyset$. Par conséquent, $J_C = \{J_P, J^+ = \emptyset, J^- = \emptyset\}$ est un support coordinateur du problème.
Construction du pseudo-plan $\kappa = (\kappa_B, \kappa_S)$:
On a $J_{NN} = \emptyset \Longrightarrow M(J_S, J_{NN}) = 0$ et $\delta = \delta(J_P) = 0$ par construction ; alors

$$\begin{aligned} \kappa_S &= -M_S^{-1}Z'(J_S, J_P)\left[D(J_P, J_B)A_B^{-1}b + c(J_P)\right] - M_S^{-1}M(J_S, J_{NN})\kappa_{NN} \\ &= -\begin{pmatrix} 1/4 & 1/4 \\ 1/4 & 1/2 \end{pmatrix}\begin{pmatrix} 1 & 0 & -1 & -2 \\ 0 & 1 & 4 & -1 \end{pmatrix}\left[\begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} 3 \\ 4 \end{pmatrix} + \begin{pmatrix} 2 \\ 1 \\ -3 \\ -1 \end{pmatrix}\right] = \begin{pmatrix} 3/4 \\ 13/4 \end{pmatrix}, \\ \kappa_B &= A_B^{-1}(b - A_N\kappa_N) = \begin{pmatrix} 3 \\ 4 \end{pmatrix} - \begin{pmatrix} 1 & -4 \\ 2 & 1 \end{pmatrix}\begin{pmatrix} 3/4 \\ 13/4 \end{pmatrix} = \begin{pmatrix} 61/4 \\ -3/4 \end{pmatrix}. \end{aligned}$$

Pour le vecteur y, on a :

$$\delta(J_B) = 0 = D(J_B, J)\kappa + c_B - A'_B y \Longrightarrow y = [A'_B]^{-1}(D(J_B, J)\kappa + c_B) = \begin{pmatrix} -3 \\ -1 \end{pmatrix}.$$

Itération 1 :
Le critère d'optimalité n'étant pas vérifié pour l'indice $j_1 = 3 \in J_B$. Construisons alors un nouveau coplan de support $\{\bar{\delta}, \bar{J}_C\}$ et son pseudo-plan correspondant $\bar{\kappa}$ de la manière suivante :

$$\bar{\delta} = \delta + \sigma^0 t,\ \bar{\kappa} = \kappa + \sigma^0 l.$$

On a : $\alpha_3 = \kappa_3 - d_3^+ = \dfrac{61}{4} - 5 = \dfrac{41}{4}$, puisque $\delta_3 = 0$ et $\kappa_3 > 5$.
• Calcul des directions d'amélioration t et l :
$t_3 = -\text{sign}\alpha_3 = -1, t_1 = t_2 = t_4 = 0,$

$$l_S = -M_S^{-1} Z' t = \begin{pmatrix} 1/4 & 1/4 \\ 1/4 & 1/2 \end{pmatrix} \begin{pmatrix} 1 & 0 & -1 & -2 \\ 0 & 1 & 4 & -1 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ -1 \\ 0 \end{pmatrix} = \begin{pmatrix} -3/4 \\ -7/4 \end{pmatrix},$$

$$l = Z l_S = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ -1 & 4 \\ -2 & -1 \end{pmatrix} \begin{pmatrix} -3/4 \\ -7/4 \end{pmatrix} = \begin{pmatrix} -3/4 \\ -7/4 \\ -25/4 \\ 3/4 \end{pmatrix}.$$

• Calcul du pas optimal σ^0 :
$J_{NN} = \emptyset$, alors $\sigma_{j_0} = \infty$;
$\sigma_{j_1} = -\dfrac{\alpha_3}{l_3} = -\dfrac{-41}{4} \times \dfrac{4}{25} = \dfrac{41}{25}$, donc $\sigma^0 = \min\{\sigma_{j_1}, \sigma_{j_0}\} = \sigma_{j_1} = \dfrac{41}{25}$.
Le nouveau pseudo-plan est :

$$\bar{\kappa} = \begin{pmatrix} 3/4 \\ 13/4 \\ 61/4 \\ -3/4 \end{pmatrix} + \frac{41}{25} \begin{pmatrix} -3/4 \\ -7/4 \\ -25/4 \\ 13/4 \end{pmatrix} = \begin{pmatrix} -12/25 \\ 19/50 \\ 5 \\ 229/50 \end{pmatrix}.$$

$\bar{\delta}_3 = \delta_3 + \sigma^0 t_3 = -41/25$, alors $\bar{\delta} = (0, 0, -41/25, 0)'$.
• Changement du support :
On a $\sigma^0 = \sigma_{j_1}$, $j_1 = 3 \in J_B$, alors le nouveau support coordinateur $\bar{J}_C = \{\bar{J}_B, \bar{J}_S, \bar{J}^+, \bar{J}^-\}$ est obtenu en posant :

$$\bar{J}_B = (J_B \backslash j_1) \cup j_*,\ \bar{J}_S = J_S \backslash j_*, \text{ où } j_* \in J_S \text{ et vérifie } z_{j_1 j_*} \neq 0.$$

On choisit par exemple $j_* = 2$, car $z_{32} = 4 \neq 0$; alors

$$\bar{J}_B = (J_B \backslash 3) \cup 2 = \{2, 4\},\ \bar{J}_S = J_S \backslash 2 = \{1\},\ \bar{J}^- = \{3\},\ \bar{J}^+ = \emptyset.$$

Le coplan de support coordinateur $\{\bar{\delta}, \bar{J}_C\}$ vérifie les relations d'optimalité (15), alors le plan optimal pour le problème (17) est :

$$x^0 = \bar{\kappa} = (-12/25, 19/50, 5, 229/50)',\ \text{avec } F(x^0) = -911/50.$$

5 Conclusion

Dans ce papier, nous avons présenté la méthode duale de support pour la résolution des problèmes quadratiques à variables bornées. L'avantage de cette méthode réside dont le fait qu'elle traite les problèmes tels qu'ils se présentent, sans chercher à les transformer, ce qui engendre un gain en espace mémoire et en temps de calcul sur machine. Elle permet aussi, à l'instar de celle de Boland [12], de traiter les problèmes convexes d'une manière générale, contrairement aux autres méthodes qui ne considèrent que le cas strictement convexe [14,13].

References

1. R. Gabassov, F. M. Kirillova and V. M. Raketsky. On methods for solving the general problem of convex quadratic programming. Soviet. Math. Dokl. **23** (1981) 653–657.
2. R. Gabassov and F. M. Kirillova and O. I. Kostyukova. Solution of linear quadratic extremal problems. Soviet. Math. Dokl. **31** (1985) 99–103.
3. R. Gabassov, F. M. Kirillova, V. M. Raketsky and O.I. Kostyukova. Constructive methods of optimization, Volume 4 : Convex Problems. University Press, Minsk (1987).
4. W. S. Dorn : Duality in quadratic programming. Quarterly of Applied Mathematics **18** (1960) 155–162.
5. M. O. Bibi. Support method for solving a linear quadratic problem with polyhedral constraints on control. Optimization **37** (1996) 139–149.
6. M. O. Bibi and A. Faradji. Méthode duale pour la minimisation d'une fonctionnelle quadratique convexe avec contraintes simples. Résumés des communications de la $4^{\text{ème}}$ rencontre de Recherche Opérationnelle, U.S.T.H.B, Alger (1996) 78–79.
7. M. O. Bibi and N. Ikheneche : Optimisation par la méthode adaptée d'un problème linéaire-quadratique convexe à variables bornées. Actes du Colloque international MSS'4, U.S.T.H.B, Alger (2004) 1–6.
8. B. Brahmi : Méthodes primale et duale pour la résolution des problèmes de programmation quadratique convexe. Mémoire de Magister, Université de Béjaîa (2007).
9. R. Fletcher. *Practical Methods of Optimization.* J. Wiley and Sons, Chichester, England, second edition (1987).
10. P. E. Gill, W. Murray, and M. H. Wright. *Practical Optimization*. Academic Press Inc, London (1981).
11. J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer-Verlag, New York (1999).
12. N. L. Boland. A dual-active-set algorithm for positive semi-definite quadratic programming. *Mathematical Programming*, **78(1)**(1997)1–27.
13. D. Goldfarb and A. U. Idnani. A numerically stable dual method for solving strictly convex quadratic problems. *Mathematical Programming*, **27** (1983) 1–33.
14. C. Van de Panne and A. Whinston. The simplex and dual method for quadratic programming. *Operational Research Quarterly Journal*, **15** (1964) 355Ű-388.

Intégration d'une Classe Virtuelle Synchrone dans une Plate Forme d'e-Learning

Louiza Belkhiri, Mohamed Ramdane, Rachid Ahmed-Ouamer

Laboratoire de Recherche en Informatique LARI, Département d'Informatique
Université Mouloud Mammeri de Tizi-Ouzou, 15000 Tizi-Ouzou, Algérie
belkhiri_liza@yahoo.fr, ramdane.moh@caramail.com, ahm_r@yahoo.fr

Abstract. L'information électronique et les technologies de l'Internet ont transformé notre façon d'apprendre. Les plates formes d'e-learning offrent aujourd'hui plusieurs potentialités d'apprentissage en ligne et une grande variété de méthodes de formation à distance adaptées à la diversité des besoins des apprenants. Toutefois la salle de classe traditionnelle demeure encore l'une des meilleures formules pour apprendre et progresser. Dans ce papier, nous proposons un environnement de l'e-learning de l'informatique qui intègre le concept de classe virtuelle synchrone permettant de reproduire fidèlement le déroulement d'une séance de cours en offrant au formateur et aux apprenants la possibilité de dialoguer à distance et de communiquer en temps réel comme pendant une session de cours classique.

Mots clés. e-Learning – FAD – Classe virtuelle synchrone – Visioconférence – Streaming – Multimédia.

1 Introduction

Actuellement les environnements de Formation A Distance (FAD) offrent le plus souvent des opportunités basées sur des échanges asynchrones (le tuteur-formateur intervient en différé à une demande de l'apprenant). Les interactions entre apprenants ou entre formateur et apprenants se font par le biais d'outils de communication asynchrone (courrier électronique, forum de discussion, foire aux questions, etc.). Toutefois la participation active des apprenants conditionne le succès et la réussite du téléapprentissage [5]. La mise en place de moyens de communication synchrone (où le formateur et l'apprenant échangent en temps réel des points de vue sur le contenu présenté) permet aux apprenants d'entretenir un niveau d'interactivité de qualité et au formateur d'adapter la forme de ses interventions à la nature des sollicitations. Cependant, les environnements de téléapprentissage synchrone sont souvent développés indépendamment des activités asynchrones [17] induisant une discontinuité dans le processus d'apprentissage, et limitant ainsi son efficacité. Notre approche consiste à intégrer dans une plate forme de FAD asynchrone une classe virtuelle synchrone pour pouvoir reproduire fidèlement le déroulement d'une séance de cours. Ce travail s'inscrit dans le cadre du développement d'environnements informatiques multimédias pour le téléapprentissage synchrone. Les études actuelles

sur l'apprentissage en ligne ont montré qu'il est difficile de concevoir un environnement unique pour toutes les formes d'activités pédagogiques synchrones [5], [17]. Celles-ci peuvent être des télé-cours (cours ou conférences en ligne), des télé-TDs (travaux dirigés et exercices en ligne), des télé-TPs (travaux pratiques en ligne), ou des télé-projets (projets nécessitant un travail collaboratif à distance). On s'intéressera ici essentiellement aux télé-cours. Les fonctionnalités premières d'une classe virtuelle synchrone sont alors la diffusion d'un cours en temps réel et la communication synchrone entre apprenants ou entre formateur et apprenants. Sa mise en œuvre repose sur les éléments suivants :

– La mise en place de l'infrastructure réseau et des techniques de diffusion de contenus multimédias qui s'appuient sur le transport de données en temps réel (RTP**,** RTCP, streaming, etc.) ; la compression et le codage de données (H323, MPEG2, MPEG4, etc.) et la synchronisation de divers flux de données (image, son, vidéo, etc.) satisfaisant les exigences de qualité nécessaires au bon suivi d'un télé-cours.

– L'intégration d'outils de communication et de collaboration (chat, partage d'applications [10], tableau blanc partagé, etc.) afin de pouvoir reproduire fidèlement le déroulement d'un cours.

Ce papier s'intéresse à la mise en œuvre de ces approches dans le contexte de l'e-learning de l'informatique. La section 2 précise les techniques de diffusion de contenus multimédias utilisées. Dans la section 3 est abordée la communication multimédia synchrone. L'environnement développé pour le téléapprentissage synchrone est décrit dans la section 4. La mise en œuvre de la classe virtuelle synchrone est ici particulièrement détaillée. La section 5 est relative aux tests et expérimentations effectués. La dernière section fait la synthèse de cette étude.

2 Techniques de Diffusion de Contenus Multimédias

En e-learning, la vidéo constitue un média privilégié. On distingue essentiellement deux modes de transmission vidéo sur Internet : la visioconférence et le streaming.

2.1 La Visioconférence [2]

La visioconférence (ou vidéoconférence) prend en charge les communications audio et vidéo entre utilisateurs distants ainsi que les échanges temps réel de divers types de données (texte, graphique, image, etc.). Les systèmes de visioconférence permettent l'acquisition du son et de la vidéo à partir de périphériques de capture (caméra et microphone). Les signaux sont codés et compressées par des codecs (Codeur-Décodeur), puis transmis à travers le réseau utilisé. De l'autre côté du réseau, les signaux sont décodés, le son et la vidéo sont reproduits et diffusés grâce à des périphériques de sortie (casques, enceintes, écrans, vidéo projecteurs). Les applications de la vidéoconférence sont nombreuses en e-learning [19], [16], [11], [14], [6]. Par exemple, dans les expériences de Schullo [22], la vidéoconférence est utilisée au sein d'une plate forme de FAD asynchrone. 80 % du temps d'apprentissage est passé en asynchrone et 20 % seulement en synchrone. Les protocoles les plus utilisés en visioconférence sont H323 et SIP.

– H323 : L'union de télécommunication internationale (ITU) a édité en 1996, la première version du standard de la communication multimédia H323, développée pour les réseaux à commutation de paquets. Celle-ci est une adaptation de la norme H320 utilisée en visioconférence pour le Réseau Numérique À intégration de Services (RNIS). Ce standard a été par la suite amélioré. La norme T120 par exemple permet l'échange de données entre applications.
– SIP (protocole d'initialisation de session) est un protocole de signalisation. Son rôle consiste à permettre l'ouverture, la modification et la libération de sessions. Celles-ci peuvent être des sessions audio, de vidéoconférence, de téléphonie (voix) ou de diffusion multimédia sur IP essentiellement.

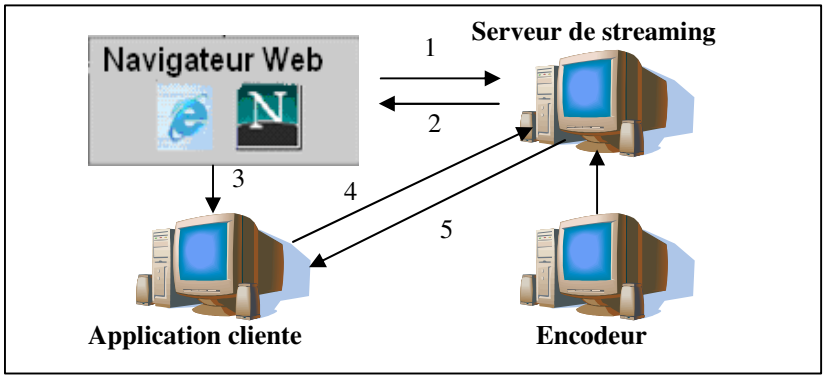
Les applications de la visioconférence dans les réseaux IP se sont développées au milieu des années 90. Les plus populaires sont probablement les programmes de messagerie instantanée (Yahoo Messenger, MSN Messenger, ivisit) [13]. Deux technologies standardisées sont alors apparues. Celles qui sont basées sur la norme H323 [18] et les outils du Mbone [7] :
– Les techniques basées sur la norme H323 [18]. Microsoft NetMeeting développé depuis 1996, demeure la technique H323 la plus célèbre pour la communication synchrone sur Internet. De nombreux environnements de FAD sont construits autour de Microsoft NetMeeting comme par exemple Speaker Tool.
– Les outils du Mbone [7]. Le Multicast Backbone est une infrastructure réseau qui permet de diffuser des informations à des groupes d'abonnés tout en optimisant l'utilisation du réseau. Les protocoles utilisés sont basés sur l'IP multicast et sont bien adaptés aux applications de visioconférence. Ce réseau offre un ensemble d'outils open source pour les communications audio et vidéo sur Internet. Par exemple les outils de communication audio, le Robust Audio Tool (RAT), et de communication vidéo (VIC) fonctionnent sur une gamme étendue de plates formes.

2.2 Le Streaming [9]

Le streaming ou webcasting (web diffusion) est un système informatique qui permet de diffuser en temps réel un flux audio-vidéo depuis un serveur de streaming vers un ou plusieurs clients (sans avoir besoin de télécharger un fichier avant de le visualiser). L'utilisateur devant installer préalablement un plug-in ou un client de streaming. A l'aide de son navigateur il peut ensuite le visualiser instantanément (ou presque) sur son poste de travail. Deux modes de diffusion sont possibles sur Internet : en direct ou à la demande. Dans le mode direct, l'utilisateur se connecte à un flux en cours qui est transmis de la même manière à tous les clients. Mais pour la vidéo à la demande (VOD), la requête est traitée uniquement pour l'usager qui en fait la demande. Plusieurs environnements de FAD synchrone ont recours au streaming pour la diffusion du son et de la vidéo sur les réseaux IP [5], [17], [16], [6], [9] conformément au schéma décrit par la figure 1.

Les spécificités des protocoles de streaming utilisés pour la transmission de données dans les réseaux TCP-IP (Internet, intranet) sont la qualité de service "best effort" (les paquets de données peuvent ne pas arriver), la transmission sans connexion "datagramme" (les paquets de données peuvent arriver dans le désordre), et la variation du temps de latence " gigue" (les paquets de données n'ont pas le même

temps de transmission – dû au trafic – d'où une arrivée irrégulière. Pour pallier à ces problèmes, des protocoles spécifiques (RTP, RTCP, RTSP, MMS, etc.) ont été implémentés et la technique de streaming réadaptée : le flux multimédia arrivant au récepteur n'est pas immédiatement lu mais mis dans une mémoire tampon ("bufferisation"). Le lecteur multimédia entame sa lecture dans la mémoire tampon quelques secondes plus tard, ce qui amène un effet compensatoire au problème de la gigue. Ces protocoles permettent aussi le "bandwidth negociation" (meilleure qualité possible selon le type de connexion utilisé). Dans ce cas, plusieurs encodages sont possibles. Le fichier transmis et celui qui correspond le mieux à la bande passante du client [16].



1. Un visiteur d'une page web clique sur un lien pour une diffusion de médias (streaming media). Ceci lance une demande au serveur de streaming.
2. Le serveur de streaming crée un fichier "metafile" qui est transmis au visiteur de la page web.
3. Le navigateur télécharge ce fichier qui est alors transmis à l'application multimédia du client.
4. L'application multimédia cliente lit le lien du fichier "metafile" et produit une requête directement au serveur de streaming.
5. Le serveur de streaming déverse en mode continu (stream) le fichier de la présentation à l'application multimédia du client qui l'affiche.

Fig. 1. Vue d'ensemble du service de streaming

Il existe actuellement trois principaux systèmes de streaming : Real Networks, Microsoft Windows Media (MWM) et Apple QuickTime. Real Networks est sans doute le plus ancien et reste aujourd'hui le système le plus utilisé car il est multi plates formes. En outre il permet par exemple de visualiser des clips SMIL (Synchronized Multimedia Integration Language). Il y a aussi les logiciels libres suivants : VLC "Vidéo LAN Client" (distribué sous licence GNU pouvant être exploité pour le développement d'application de streaming), et JMF "Java Media Framework" (qui fournit des composants pour construire des vidéo-streamers).

Parmi les objectifs visés par notre environnement de téléapprentissage synchrone, la simplification des contraintes (matérielles et logicielles) induites à son utilisation est primordiale. De plus, dans le contexte d'un télé-cours, il ne doit pas y avoir de restrictions sur le nombre d'apprenants engagés dans une activité d'apprentissage. La visioconférence permet des échanges synchrones bidirectionnels entre usagers mais

elle reste peu adaptée lorsque le nombre d'utilisateurs augmente, nécessitant alors une large bande passante. En outre, elle requiert d'équiper le poste client avec des périphériques d'acquisition du son (microphone) et de la vidéo (caméra). Notre plate forme d'e-learning doit pouvoir prendre en compte une population d'apprenants possédant des connexions réseaux différentes (LAN, modem 56k, RTC, ADSL, etc.). Elle doit permettre le bon suivi des activités d'apprentissage synchrone quelle que soit la connexion utilisée. Le streaming est donc mieux adapté à notre application.

3 Communication Multimédia Synchrone

La communication synchrone impose des contraintes techniques plus restrictives que celles d'une communication asynchrone. Les informations échangées dans les activités d'apprentissage synchrone sont émises en temps réel induisant des temps de réponse critiques. Les principaux outils utilisés dans les environnements de téléapprentissage synchrone sont les outils de communication informelle entre utilisateurs (le chat, la vidéoconférence, etc.), les outils de partage de documents (le tableau blanc partagé, la diffusion de clip vidéo, etc.) et les applications multi utilisateurs (le partage d'applications, l'explorateur Internet partagé, etc.). Une classification de ces outils a été établie par Baudin et peut être trouvée dans [2].

En classe virtuelle synchrone, les apprenants doivent pouvoir discuter en temps réel avec leur formateur. La question qui se pose alors est celle de savoir quelles formes doivent prendre ces communications (écrite, audio ou audiovisuelle) ?
Plusieurs facteurs ont influencé nos choix :

– Sur le plan technique : les débits des réseaux grand public ne permettent pas d'avoir des qualités de son et de vidéo satisfaisantes. Cependant une connexion par modem est suffisante pour un outil de communication textuel [12].
– Sur le plan pédagogique : les outils de communication doivent permettre aux apprenants de poser des questions et de faire des interventions pertinentes [8] ;
– Sur le plan psychopédagogique : des expériences ont montré que la communication vidéo peut parfois avoir chez certains apprenants un effet d'inhibition de l'interaction : la perspective pour un apprenant d'être filmé et de voir son image diffusée à l'ensemble des participants peut provoquer chez lui un réflexe d'autocensure [9]. En outre, les outils de communication textuelle sont bien adaptés aux apprenants timides, pensifs et hésitants [4].

Nous avons donc préféré intégrer les outils suivants dans notre plate forme :

– Le chat : il permet à plusieurs usagers de dialoguer par écrit de façon synchrone ;
– La messagerie instantanée : elle permet d'envoyer instantanément un message à un seul utilisateur et non à l'ensemble des usagers connectés ;
– Le paging (présence) : il permet à tout utilisateur de savoir qui est en ligne et de fournir des informations sur les utilisateurs présents en classe virtuelle.
– Le tableau blanc partagé [3] : il permet de travailler à plusieurs sur des données (texte ou graphique) en reprenant le concept du tableau classique sur lequel tout le monde peut lire et écrire. Le tableau blanc partagé est le plus souvent utilisé pour l'annotation (voir figure 2) : souligner un détail, entourer un objet, dessiner ou écrire à main levée, etc.

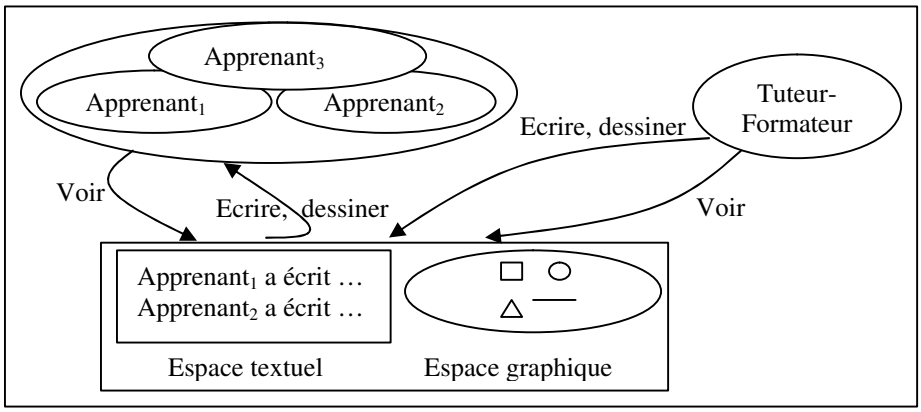


Fig. 2. Principe d'utilisation du tableau blanc partagé

4 Mise en Œuvre de la Classe Virtuelle Synchrone

Cette section décrit la mise en œuvre de la classe virtuelle synchrone développée pour l'e-learning de l'informatique. Ses architectures matérielle et logicielle sont présentées ainsi que ses fonctionnalités principales.

4.1 Architecture Matérielle de la Plate Forme d'e-Learning Développée

L'architecture matérielle de notre plate forme de FAD est décrite dans la figure 3. Outre le poste client, nous distinguons deux parties dans l'infrastructure proposée : le studio de cours et la régie de diffusion (qui constitue le point critique du système). Le studio de cours est équipé pour la production des contenus (flux) avec une orientation temps réel. Dans ce studio, le formateur dispose de deux postes : le premier appelé "live-cours", accueille et transmet (encodeur de flux écran et son) le support de cours du formateur sur lequel des annotations peuvent être faites. Il est aussi possible d'utiliser un écran interactif (tableau tactile et vidéoprojecteur). Le second est un poste de communication (interface web). Il regroupe les outils de communication entre formateur et apprenants. Un troisième poste relié à une caméra filme et encode la vidéo du formateur (encodeur de flux vidéo). La régie de diffusion effectue la récupération et le stockage en temps réel des flux venant des encodeurs, la distribution (vers les postes clients et via Internet) de contenus de natures et d'origines diverses (vidéo-cours, démonstrations filmées, etc.) en temps réel ou en temps différé ainsi que la liaison avec la plate forme d'e-learning jouant le rôle de portail d'accès (gestion de cursus personnalisés, gestion des apprenants, espace de travail collectif, etc.).

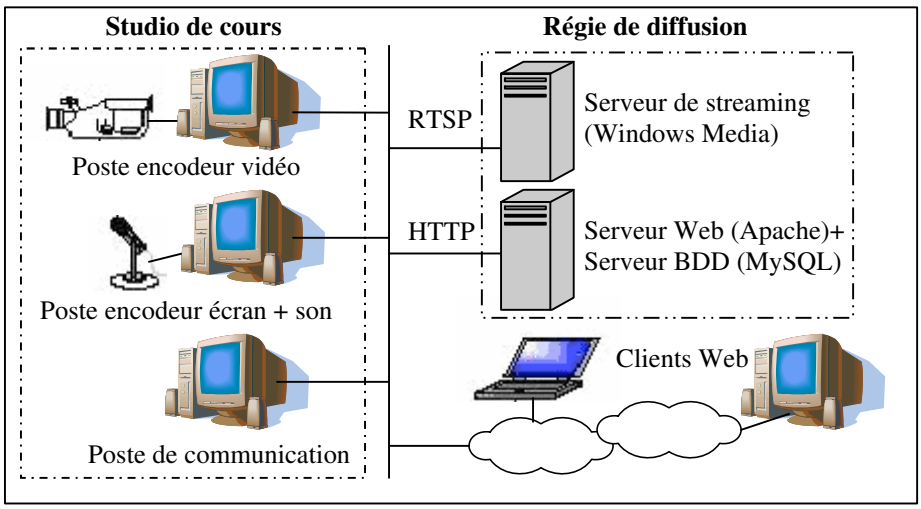


Fig. 3. Architecture matérielle de la plate forme d'e-learning développée

4.2 Architecture Logicielle de la Plate Forme d'e-Learning Développée

Un serveur web (Apache) héberge notre plate forme. Celle-ci est basée sur la technologie Hypertext Preprocessor (PHP) qui permet de mettre en place un couplage site web-base de données (MySQL). Le tableau blanc partagé (cf. figure 4) est fondé sur une architecture client/serveur. L'application serveur joue le rôle de relais pour les divers clients afin de pouvoir communiquer entre eux. Un client est une application lancée par un utilisateur pour se connecter au tableau blanc partagé via le serveur. Il est implémenté par une technique de type "applet Java" qui est téléchargée dynamiquement par le navigateur lorsque l'apprenant clique sur le lien correspondant. Pour la diffusion du cours, du son et de la vidéo du formateur, notre environnement utilise le streaming à partir de la technologie MWM dont la chaîne d'acquisition est basée principalement sur l'encodeur, le serveur MWM et le client :

– L'encodeur : il s'agit du poste qui réalise l'acquisition des données depuis un périphérique tel qu'une caméra. Il est situé près d'une source de données. Il encode les informations suivant un profil particulier qui définit la taille de la vidéo capturée ou encore le débit nécessaire à la diffusion.
– Le serveur MWM : il permet de centraliser les flux venant des encodeurs et de retransmettre ces flux en temps réel ou en temps différé.
– Le client (lecteur vidéo) : il constitue la partie cliente de l'infrastructure. Ce lecteur effectue les demandes vers le serveur web pour recevoir et afficher les flux. Il s'adapte au format des vidéos reçues en téléchargeant automatiquement le codec (programme de compression / décompression) adapté au profil utilisé sur l'encodeur. Le serveur MWM peut gérer plusieurs types de flux suivant différents modes (flux en temps réel ou en temps différé, transmission en unicast ou en multicast) ainsi que la gestion de la transmission en UDP, TCP avec le protocole propriétaire MMS. Il est

aussi possible d'utiliser le protocole HTTP permettant l'encapsulation de données vidéo dans des trames http afin d'éviter éventuellement le problème de traversée des pare-feux qui bloqueraient les flux de type MMS utilisant un port non conventionnel.

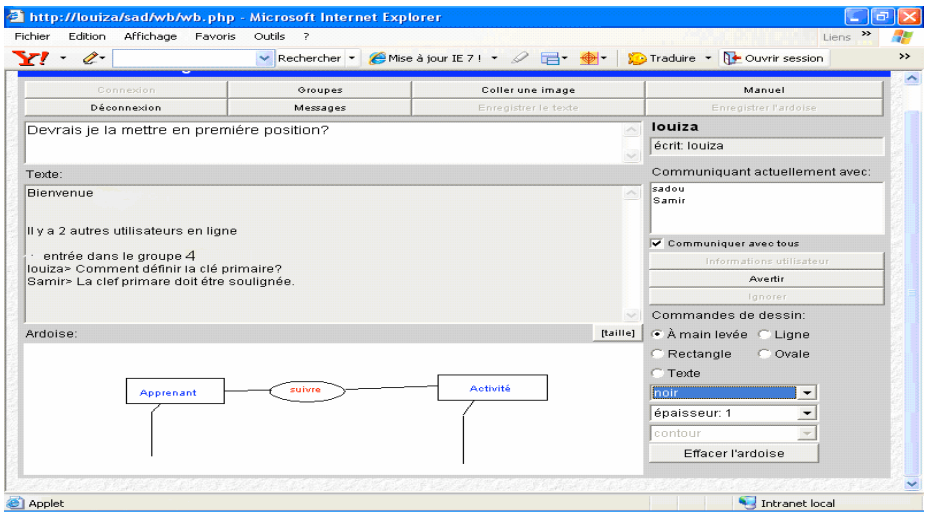


Fig. 4. Exemple d'utilisation du tableau blanc partagé

4.3 Services, Outils et Interfaces

La plate forme d'e-learning comporte trois interfaces principales :
– Une interface apprenant (figure 5) : elle comprend l'accès aux éléments du cours et aux différents outils de la plate forme (espace de travail apprenant, accès aux activités pédagogiques, etc.).
– Une interface formateur : elle permet le lancement de cours en temps réel ainsi que l'échange d'informations et le dialogue entre formateur et apprenants.
– Une interface auteur-administrateur : elle permet à l'auteur-concepteur le développement de contenus pédagogiques et à l'administrateur de la plate forme, la gestion des données et des informations administratives (gestion des inscriptions, droits d'accès, création et affectation de groupes, etc.).

La description complète des fonctionnalités de la plate forme d'e-learning développée peut être trouvée dans [20] et dans [21]. Ce papier est consacré essentiellement au volet synchrone de cet environnement de FAD.

Côté Formateur. En studio de cours, le formateur dispose d'un poste sur lequel il visualise son support de cours (un document Powerpoint par exemple). Dans le cadre de son activité pédagogique, le formateur peut utiliser le poste "live-cours" pour lancer une application spécifique. Les apprenants distants peuvent suivre les actions réalisées par leur formateur via le poste relié à la caméra. Celle-ci capture en temps réel la vidéo du formateur. Ceci permet au formateur de dispenser son cours d'une

façon plus autonome. Enfin, le formateur dispose d'un poste de communication pour dialoguer avec les apprenants à l'aide d'outils de communication (chat, mail, tableau blanc partagé, etc.).

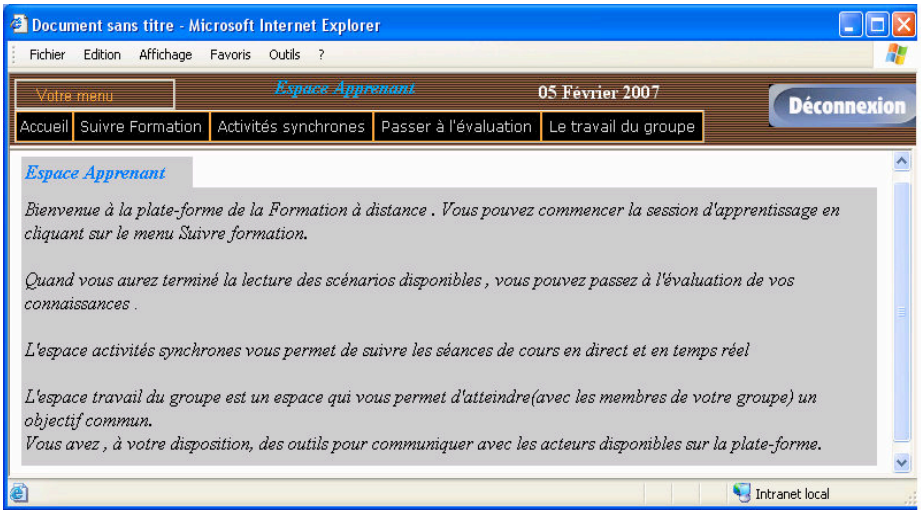


Fig. 5. Interface de l'espace apprenant

Côté Apprenant. Sur les postes clients, l'objectif est d'imposer un minimum de contraintes (environnement full web) sur le matériel et le logiciel utilisés. Par l'intermédiaire d'un navigateur, l'environnement permet la visualisation de la vidéo du formateur, celle de son support de cours de type diaporama ou autre application partagée et l'interactivité pendant les séances en live (questions, réponses, utilisation du tableau blanc partagé, etc.), comme il est indiqué sur la figure 6. Le module de visualisation de vidéos intégré dans l'interface apprenant permet la lecture en temps réel ou en temps différé de flux vidéo, la synchronisation et la commande de flux (lecture, pause, etc.), la mise en place de marqueurs et de déplacement assisté dans les flux ainsi que l'adaptation à la résolution des postes clients distants.

5 Tests et Evaluation

Avant d'évaluer cette classe virtuelle synchrone, il est nécessaire d'effectuer certains tests permettant de déterminer les paramètres recommandés pour l'accès au réseau et l'utilisation de l'encodeur.

Accès au Réseau et Utilisation de l'Encodeur. L'accès à la classe virtuelle synchrone requiert un poste client capable de recevoir et de traiter les données mais aussi une connexion réseau apte à assurer convenablement leur acheminement. Son expérimentation a été effectuée localement en intranet. Nous avons utilisé le logiciel

WME (Windows Media Encoder) 9.1 pour réaliser l'encodage. L'encodeur fournit des paramètres pour adapter la vidéo à sa distribution sur le réseau. Afin de minimiser la charge du réseau, la qualité de la transmission peut être réduite et codée dans des formats différents.

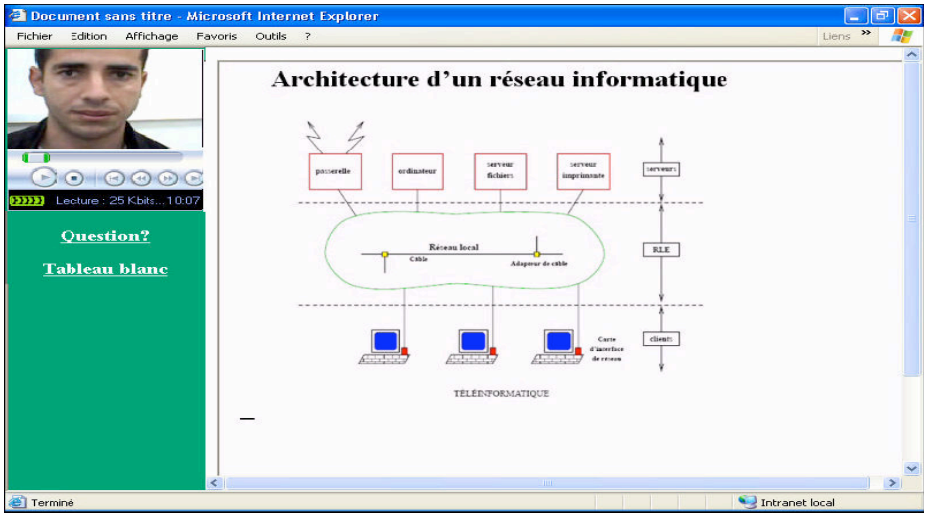


Fig. 6. Exemple d'interface vue par un apprenant lors d'un télé-cours synchrone

WME fournit des profils prédéfinis pour différentes bandes passantes. Nous avons fait varier les paramètres de l'encodeur vidéo et de l'encodeur du cours / son pour obtenir diverses qualités de transmission :

– Pour l'encodage de la vidéo, WME utilise les résolutions par défaut suivantes : 160x120, 176x144 (QCIF), 320x240, 352x288 (CIF) et 640x480. Pour notre application, nous avons constaté que QCIF est la mieux adaptée à la transmission.

– Concernant le support de cours, nous avons utilisé le format spécifique (full screen) car l'encodeur supporte la capture d'écran comme périphérique d'entrée.

– Quant au son, un encodage bas débit suffit pour obtenir une qualité acceptable. Nous avons préféré de l'encodage à plusieurs débits (Multiple Bit Rates).

Dans le contexte d'une activité de télé-cours, la vidéo du formateur n'est pas un média principal. Elle ne véhicule pas d'échanges informatifs mais plutôt des informations comportementales comme l'émotivité ou l'émotion. Les apprenants peuvent s'ils le désirent interrompre la réception de la vidéo du tueur-formateur. Cependant celle-ci contribue à rompre le sentiment d'isolement qui constitue un frein principal à l'acceptation des environnements de FAD [18]. En revanche, le son est un média prépondérant : le formateur peut répondre oralement aux questions écrites, posées par les apprenants. En outre le son véhicule les commentaires du formateur en rapport avec son activité de tutorat. Il est également possible de modifier la présentation dans WME en ajoutant des commentaires (auteur, titre, etc.) qui seront directement visibles sur les postes clients.

Interface Web et Outils Intégrés. L'interface web est conviviale et facile à utiliser. L'apprenant se connecte au site et s'authentifie avec les paramètres de connexion qui lui ont été attribués par l'administrateur. Après validation, l'apprenant accède à son espace de travail comme le montre la figure 5 précédente. Un simple clic sur le lien "activités synchrones" lui permet de réceptionner directement les flux audio, vidéo et écran transmis depuis le studio de cours de la plate forme d'e-learning, et ainsi de suivre en temps réel le cours transmis comme il est indiqué sur la figure 6 précédente. Les outils intégrés sont bien adaptés à l'apprentissage synchrone [1]. En effet, les questions écrites par exemple permettent au formateur de répondre au moment qui lui convient le mieux de façon à ne pas interrompre sa présentation et de sélectionner par la suite uniquement celles qu'il juge plus pertinentes. Ceci lui permet d'éviter d'éventuelles perturbations au bon déroulement de son cours. Cependant, les activités d'apprentissage synchrone nécessitent une préparation préalable et une planification adéquate qui induisent quelque peu une charge supplémentaire aux formateurs [15].

6 Conclusion

Dans ce papier nous avons abordé la problématique temps réel du téléapprentissage. Les éléments clefs d'une solution de FAD synchrone ont été mis en évidence. Un premier résultat se trouve dans l'environnement expérimental développé pour permettre la mise en œuvre de la classe virtuelle synchrone offrant aux apprenants distants l'opportunité d'assister en direct et de suivre en temps réel des activités pédagogiques (essentiellement des télé-cours) ainsi que la possibilité de communiquer entre eux ou avec leur formateur à l'aide d'outils de communication et de collaboration via un simple navigateur web. La plate forme d'e-learning développée selon une architecture n-tiers (client-serveur Web-SGBD- serveur de streaming) est réalisée avec le langage de scripts PHP (côté serveur), JavaScript et applet java (côté client) pour la partie dynamique, et les langages HTML et XML pour la partie statique. Le langage MySQL est utilisé pour l'interrogation de la base de données sous-jacente. Les outils de communication et de collaboration synchrones intégrés, permettent un échange de points de vue simple et efficace entre apprenants ou entre apprenants et leur formateur. Cet environnement répond bien aux besoins de généricité et d'extensibilité, puisqu'il est indépendant de la nature des activités pédagogiques pouvant être utilisées. De nouvelles fonctionnalités peuvent lui être facilement ajoutées pour améliorer l'interactivité, la communication et la collaboration entre apprenants (ou entre formateur et apprenants), ou pour la mise en œuvre d'autres types d'activités d'apprentissage synchrone, individuelles ou collectives, etc. Les tests effectués localement (en intranet) sont assez satisfaisants. Notre prochaine étape consiste à enrichir le domaine d'application utilisé (quatre didacticiels : l'algorithmique, les bases de données, les systèmes d'exploitation, et la téléinformatique et six scénarios : deux pour les systèmes d'exploitation, deux pour les bases de données, un pour la téléinformatique et un autre pour l'algorithmique sont développés) [20], [21] en intégrant un espace de travail collaboratif synchrone (télé-TDs, télé-TPs et télé-projets) pour pouvoir évaluer cette plate forme de l'e-learning de l'informatique dans un environnement réel d'exploitation.

Références

1. Avouris N., Margaritis M., Komis V.: Real Time Collaborative Problem Solving: a Study on Alternative Coordination Mechanisms.IEEE Int.Conf.Adv.Learn.Tech. Athens (2003) 86–90
2. Baudin-Thomas V., Courtia J.P., Gomes R.L., Hoyos-Rivera G.J., Villemur T.: An e-Learning Collaborative Platform for Laboratory Education. 4th Int. Conf. on Inf. Tech. Based Higher Education and Training: ITHET'03, Marrakech (2003) 200–205
3. Baudin V., Raymond D., Yano Y.: Déploiement et Evaluation d'un Environnement de Travail Coopératif dans un Scénario d'Apprentissage de Langue. Int. Conf. on New Tech. for Distributed Sys., Toulouse (2006) 105–116
4. Berge Z.L.: Computer Conferencing and On-Line Classroom. Int. J. Edu. Telecom. Vol. 3 (1997)
5. Billois G., Bernard J., Ramel J.Y., Nait-Abdesselam F., Millios S.: IOL : Une Plate Forme Multimédia pour la Diffusion d'Enseignements Synchrones sur Internet. EduTice (2002)
6. Bisiaux G., Courtaud D., Monget M., Prevost J., Sol G.: Projet DIM : Téléenseignement pour les DESS Nouvelles Technologies. JRES, Lille (2001) 433–446
7. Bisiaux G., Rapacchi B.: Déploiement de la Visioconférence IP dans un Etablissement : Etat de l'Art et Evolution des Protocoles. JRES (2003)
8. Chanier T., Vetter A.: Multimodalité et Expression en Langue Etrangère dans une Plate Forme Audio-Synchrone. Revue Alsic, Vol. 9, juin (2006)
9. Colin J.F.: Conception d'un Outil Synchrone d'Enseignement à Distance. EduTice (2004).
10. Colin J.F.: Intégration de l'Outil de Partage TuttelShare® dans la Plate Forme Synchrone Tuttelnet®. EduTice (2006)
11. Courdier R.: Briser l'Isolement grâce aux Nouvelles Technologies : un Modèle d'Apprentissage à Distance en Université Reposant sur Six Années d'Expérience. Int. Conf. on Open & Online Learning : ICOOL, France (2003)
12. George S.: Apprentissage Collectif à Distance, SPLACH. Thèse de Doctorat en Informatique. Univ. LIUM, Maine (2001)
13. Goyard E.: Etat des Lieux des Ressources Multimédias Hors Ligne et en Ligne dans le Champ des Outils Synchrones pour la FOAD. Algora: www.algora.org. (2006)
14. Harvey L., Beaulieu M., Demers B., Proux J.: L'Enseignement Synchrone Multi Médiatisé à Distance : Vidéoconférence, Internet ou de Retour à la Classe Régulière ? Revue du conseil québécois de la formation à distance. (1999)
15. Osguthorpe R.T., Graham C.R.: Blended Learning Environments. The Quarterly Review of Distance Education. Vol. 4, 3 (2003) 227–233
16. Piot B., Ajirent S., David L.: Le Téléenseignement dans le Réseau du CNAM et la QoS Appliquée à la Visioconférence. JRES (2005)
17. Raymond D. : Modélisation des Communications dans des Activités d'Apprentissage à Distance Synchrones. 6ème Congrès EDSYS, Toulouse (2005) 8p.
18. Raymond D.: Synchronous Environments for Distance Learning: Combining the Network and Collaborative Approaches. Thèse de Doctorat en Ingénierie des Sciences de l'Information et Systèmes Intelligents. Univ. de Tokushima, Japon (2006)
19. Raymond D., Yano Y., Kanenishi K., Matsuura K.: Feasibility of Videoconference-Based Lectures over the Internet. J. of Learning Tech., Vol 2, 1 (2006)
20. Sadou M., Ahmed-Ouamer R.: Support aux Activités Individuelles et Collectives en e-Learning. Colloque CEMAFORAD 3, Sousse (2006) 12 p.
21. Sadou M., Ahmed-Ouamer R.:Support au Tutorat en Ligne. Colloque COSI'06, Alger (2006) 162–173
22. Schullo S., Siekmann S., Szydlo S.: Synchronous Distance Education Systems, Choosing the Right Solution. World Conf. on e-Learning in Corporate, Government, Healthcare, and Higher Education: e-Learn, AACE, Phoenix, 2421–2427

Performances evaluation of the Labeled Neuro-Fuzzy Classifier

Mohamed NEMISSI[1], Hamid SERIDI[1,2] and Mohamed Zine AISSAOUI[3]

[1]LAIG, Université de Guelma, B. P. 401, Guelma 24000, Algérie
[2]CResTIC, Université de Reims, B. P. 1035, 51687, Reims Cedex 02, France
[3]LGEG, Université de Guelma, B. P. 401, Guelma 24000, Algérie
{nemissi_m, seridi, aissaouizine}@yahoo.fr

Abstract. The LNFC (Labeled Neuro-Fuzzy Classifier) is a classification model proposed in [1] [2], its implementation aims to improve performances of the conventional NFC (Neuro-Fuzzy Classifier). This paper evaluates performances of this classifier by comparison, in one hand, with conventional NFC and, in the other hand, with two neural-classifiers: the RBF nets (Radial basis function networks) and the MLP (Multi Layered Perceptron). Iris, human thigh and texture databases will be used for the comparison. Thus, tests will be carried out with each classifier to evaluate their capacities of learning and classification. According to the obtained results, we will discuss the strengths and weaknesses of the LNFC compared to these classifiers.

Keywords: Pattern recognition, classification, neural networks, RBF neural networks, Neuro-Fuzzy Systems.

1 Introduction

In the past few years, ANNs (Artificial Neural Networks) have been widely used in several application of the pattern recognition. They have been employed as powerful classifiers for the reason of their capacities of learning and generalizing.

ANNs and FIS (Fuzzy Inference Systems) are appropriate to describe complex systems where it is difficult to give mathematical description. Moreover, a combination of these complementary methods allows having more robust systems [6][7][8][9]. For example, ANNs are not interpretable, so they are not able to represent knowledge explicitly while a fuzzy system can do it by fuzzy if-then rules [6][7]. Furthermore, Implementation of FIS necessitates tuning of the membership function parameters that can be automatically updated in the case of Neuro-Fuzzy Systems [6].

Generally, Neuro-Fuzzy Systems allows, not only, to combine advantages of ANNs and FIS, but also, to compensate their inconveniences as it indicates in table 1.

Table 1. Advantages and disadvantages of ANN and FIS.

	Advantage	Disadvantage
ANN	Parallel computing Capacity of generalization Self-adaptation	Black box Lack of initialization techniques
FIS	Possibility to use a prior knowledge	Lack of training techniques

The BP (Back propagation) algorithm is a useful algorithm in many applications. It is widely used for training the ANNs and the Neuro-Fuzzy Classifiers, but its convergence rate is relatively slow [4][5] and it is unreliable algorithm [3]. In order to improve performances of the BP, several approach have been proposed. The labeled classification is different from these methods in the sense that our approach aims to improve the training process by changing the representation of examples instead of modifying the training algorithm.

The implementation of LNFC aims to improve performances of the conventional NFC by the use of the labeled classification. The principle scheme of this approach is to add a set of labels to all training examples; tests will be then carried out with each of these labels to classify a new example [1]. The simplicity of implementation is one advantage of this approach because no modifications in the training algorithms are involved in the system.

In this work, the LNFC is compared with the NFC and with two architectures of ANNs: MLP and RBF nets. These two models provide techniques of approximating arbitrary non-linear functional mapping between multi-dimensional spaces [3]. In section 2, we describe the labeled classification. In sections 3 and 4, we present the NFC and the LNFC respectively. We discuss the obtained results on iris, human thigh and texture databases in section 5. Finally, conclusions are given in section 6.

2 The labeled classification

An important reason of the BP slowness is the saturation behavior of the activation function used in different layer [4]. In fact, when a sigmoid has a slope near a zero, a weight point may enter the saturation region of the weight space [3]. In such situation, the weight increment remains little even if the error is relatively large. The basic idea of our approach is to make the training example linearly separable by adding an extra feature (labels). The labels must be identical for examples belongings to the same class to ensure the linear separation and reduce the possibility of entering the saturation region. After the training process, every test example will be classified according to the classifier's outputs with all labels (fig. 1).

Recall that a conventional classifier maps any input vector $X(x_1\ x_2\ \ldots\ x_N)$ into an output vector $Z(z_1\ z_2\ \ldots\ z_K)$ corresponding to the class C_i of the example represented by X. Fig. 1 shows a labeled classifier with two input and three classes. It is based on the adding of labels at the input of the used classifier and giving a decision according to the classifier outputs.

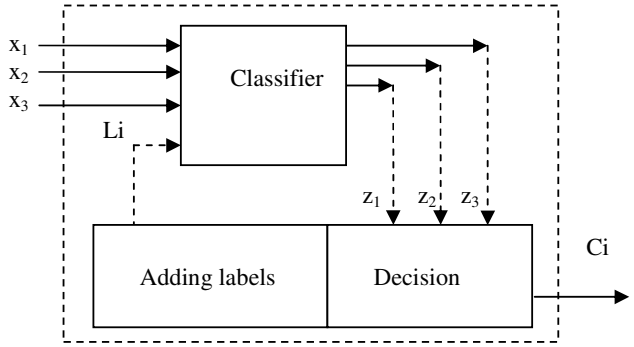


Fig. 1. Labeled classifier

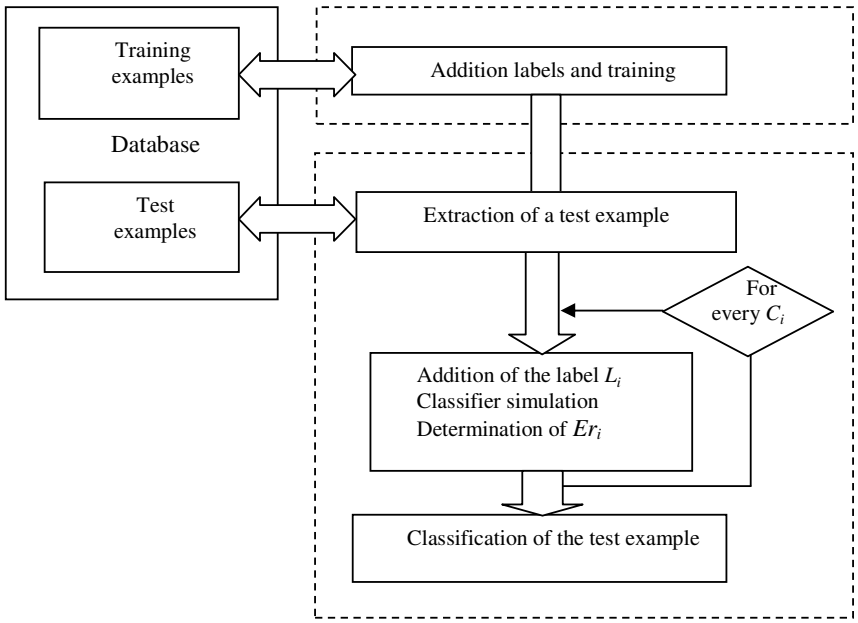


Fig. 2. Scheme of the Labeled classification

Fig. 2 shows the scheme of the labeled classification. Each class C_i corresponds to a label L_i. Then, every new example (X) will be classified according to the following decision rule:

$$X \in C_i \text{ if } Er_i(X) = \min \{Er_1(X), Er_2(X), \ldots Er_K(X)\}$$

Where Er_i is the sum-squared error between the target output $T^{(i)}$ $(t_1\ t_2\ \ldots t_K)$ corresponding to the class C_i and the classifier output $Z^{(i)}$ $(z_1\ z_2\ \ldots z_K)$ using the label L_i. Er_i is defined by:

$$Er_i = \sum_{k=1}^{K} (t_k - z_k)^2$$

3 Classification using NFC

The implementation of Neuro-Fuzzy Systems aims to combine proprieties and advantages of ANNs and FIS. In these systems every layer of ANN performs a different function of a FIS: Fuzzification, Inference and Defuzzification. The NFC (Neuro-Fuzzy Classifier) [6] has a Neuro-Fuzzy architecture that can incorporate in its structure fuzzy if-then rule of the form:

If x_1 is 'small' and x_2 is 'big' then X belongs to C_k

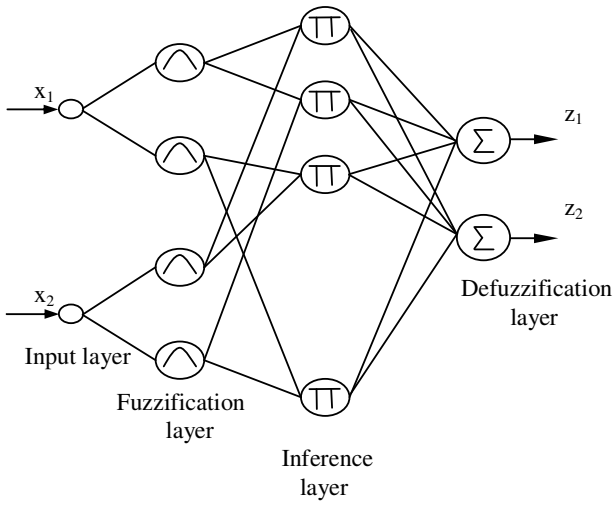


Fig. 3. Neuro-Fuzzy Classifier with two inputs, four rules and two output.

The NFC presents the advantage of using a priori knowledge to initialize its parameters. This propriety allows this classifier to begin the training of an

initialization point not far away from the optimal one [6]. Moreover, parameters gotten after the training can be transformed in a structure based on the fuzzy if-then rules. Fig. 3 illustrates the Neuro-Fuzzy Classifier. This last is composed of three layers permitting to establish a system of classification based on the fuzzy rules.

In the first layer, every neuron corresponds to a linguistic variable. Its outputs have the form:

$$s_i = \mu_{ni}(x_n)$$

Where μ_{ni} is the membership function of A_i and x_n is the n^{th} input. We use Gaussian function, so μ_{ni} has the following form:

$$\mu_{ni}(x_n) = \exp\left(-\left(\frac{x_n - c_{ni}}{\sigma_{ni}}\right)^2\right)$$

Where c_{Ai} and a_{Ai} are the parameters corresponding to A_i

Neurons of the second layer send the product of the incoming signals. The output of the m^{th} neuron has the form:

$$y_m = \prod_{n=1}^{N} {}_{ni}(x_n)$$

Every neuron of the third layer corresponds to a class. The output of the j^{th} neuron has the form:

$$z_k = \frac{\sum_m \left(\prod_n {}_{ni}(x_n)\right) w_{mk}}{\sum_m \left(\prod_n {}_{ni}(x_n)\right)}$$

In the original version of Jang [6], the output of the j^{th} neuron has the form:

$$z_j = h\left(\sum_{m=1}^{M} w_{mk} \cdot y_m\right)$$

Where h(.) is the sigmoid function and M is the number of the hidden layer neurons.

The back propagation algorithm is widely used for Neuro-Fuzzy Systems training, for example in [1] [2] [6] [7] [8] [9]. The adaptation task is to minimize the total sum-squared error E between the classifier outputs and the target outputs. E is defined over all (Q) training examples and all (K) outputs values as:

$$E = \sum_{q=1}^{Q} \sum_{k=1}^{K} (t_k^{(q)} - z_k^{(q)})^2$$

The adaptation expressions of the weight w_{mk} at the iteration (i+1) is:

$$w_{mk}^{(i+1)} = w_{mk}^{(i)} - \eta_w \frac{\partial E^{(i)}}{\partial w_{mk}}$$

The adaptation expressions of membership function parameters at the iteration (i+1) are:

$$C_{ni}^{(i+1)} = C_{ni}^{(r)} - \eta_c \frac{\partial E^{(ir)}}{\partial C_{ni}}$$

$$\sigma_{ni}^{(r+1)} = \sigma_{ni}^{(r)} - \eta_\sigma \frac{\partial E}{\partial \sigma_{ni}^{(r)}}$$

Where: η_c and η_σ are the steps gains of the center and the width of membership function respectively.

4 The LNFC: Labeled Neuro-Fuzzy Classifier

The conception of the LNFC aims to exploit and improve proprieties of the NFC [1] [2]. The use of LNFC leads to replace rules of the form:

$$\text{If } x_1 \text{ is } A_1 \text{ and } x_2 \text{ is } B_1 \text{ then } X \in C_k$$

By rules of the form:

$$\text{If } x_1 \text{ is } A_1 \text{ and } x_2 \text{ is } B_1 \text{ and } x_3 \text{ is } L_1 \text{ then } X \in C_k$$

4.1 Architecture

The labeled classification consists essentially in adding labels to all training examples. Consequently, in adding a neuron at the first layer and K neurons at the second (K is the number of classes). Every neuron added to the second layer corresponds to one label.

Fig. 4 shows an example of LNFC with two input variables (x_1 x_2) and two output variables (z_1 z_2). Every input is represented by two linguistics variables. : A_1 and A_2 are the linguistic variables characterized by the membership's functions μ_{A1} and μ_{A2}; B_1 and B2 are characterized by μ_{B1} and μ_{B2}; L_1 and L_2 are respectively the corresponding labels to the C_1 and C_2 characterized by μ_{L1} and μ_{L2}.

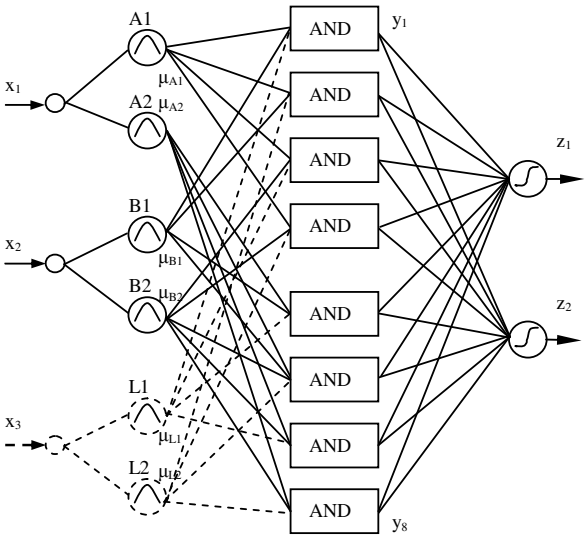


Fig. 4. LNFC with tow labels, two inputs, eight rules and two outputs.

Where L_i is the label and μ_{Li} is the parameter corresponding to L_i

4.2 Training

Our model is trained using the BP algorithm. It is performed without tuning the membership functions, which allows:

1. Obtaining a simplified training process because only the output weights are updated.
2. Keeping the original linguistic meaning of the membership functions.
3. Changing the T-norme operator (used in neurons of the third layer) without changing the training algorithm.
4. Changing the membership functions without changing the training algorithm.

4.3 Choice of labels

The premises of the Fuzzy rules established by the third layer depend on the membership functions of labels instead by the labels themselves. That is to say, the classification performances are not directly influenced by the labels values. We suggest choosing μ_{Li} (Li)=1 and μ_{Li} (Lj) = φ with φ ∈ [0.5, 1[

5 Test and experiments

5.1 Iris database classification

To appreciate the performances of the proposed model, tests are carried out on the Iris database using the NFC and LNFC. In both cases, we used three linguistic variables (fig. 5): Small, Medium and Big.

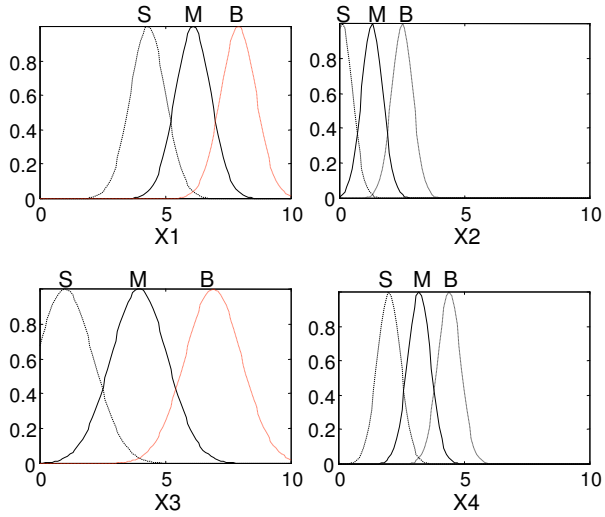


Fig. 5. Memberships functions of Iris database features.

Fig. 6 shows the evolution of the classification rate during the training stage of the NFC and the LNFC. Both models are initialized by the same weights and the training parameters are the same. The used membership function of labels are $\mu_i(Li)=1$ and $\mu_i(Lj)=0.85$.

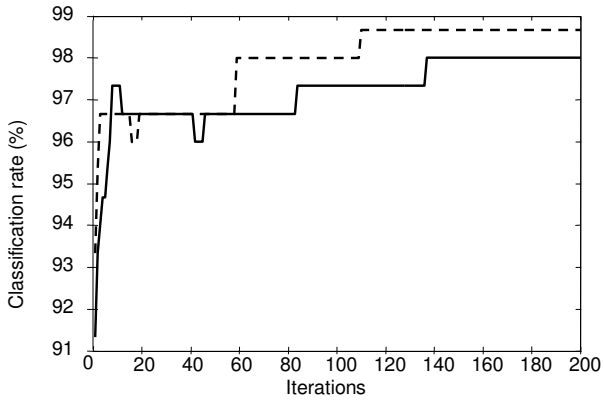


Fig. 6. Evolution of the classification rate
——— NFC
- - - - LNFC

5.2 Human thigh database classification

The image of fig. 7 is acquired by cryosection color photography. This image is put under format TIFF. It has a size of 670*415 pixels. A manual classification was made by an expert and four components were identified (grease, bone, marrow and muscle). Each one of these components corresponds to a class. The obtained sample consists of 1200 pixels, 300 pixels of each class.

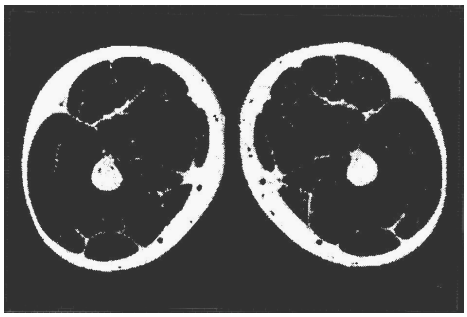


Fig. 7. Image of human thigh creyosection

The addition of components X and Y (to locate geometrical position of a pixel and to take account of its vicinity) improves the classification performances.

To evaluate generalization capacities of these classifiers, a cross validation of order 4 is used. Four databases are obtained. Each base contains 900 training examples of and 300 test examples.

As in [10], the used MLP is composed of 5 neurons at the input, 8 hidden neurons and 4 outputs neurons. The RBF network has 15 RBF-neurons obtained by the K-means method. The NFC and the LNFC are based on the use of 3 linguistic variables: Small, Medium and Big (similar to the Iris case). The used labels are: $\mu_i(Li)=1$ and $\mu_i(Lj)=0.80$. In table (2), the obtained results are showed. This results are the average of the 4 data sets obtained by the cross validation.

Table 2. Results of the human database classification

MLP	Classification rate	97.67 %
	Iterations	650
RBF	Classification rate	97.58 %
	Iterations	25+20
NFC	Classification rate	97.92 %
	Iterations	15.25
LNFC	Classification rate	**98.17 %**
	Iterations	**11.25**

5.3 Texture database classification

Fig. 8 shows an image constituted of two different microtextures. A pre-treatment (calculation of different local correlations) of the initial image allows obtaining a series of 8 images. Each one is the detection result of a particular attribute. A pixel is then described by a vector of 8 attributes. Thus, the image is represented by two classes which are described by 8 files containing each one the pixel values. The obtained sample consists of 400 pixels of each class. A cross validation of order 4 allows to obtain 4 training data sets containing each one 600 pixels and 4 test data sets of 200 pixels for each one.

The used MLP is composed of 8 neurons at the input, 8 hidden neurons and 2 outputs neurons. We used an RBF network with 12 RBF-neurons obtained by the K-means method. The NFC and the LNFC are based on the use of 2 linguistic variables. The used labels are: $\mu_i(Li)=1$ and $\mu_i(Lj)=0.85$. Table (3) indicates the obtained results (the overage of 4 data sets).

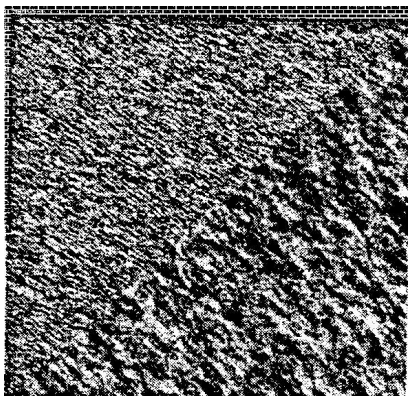


Fig. 8. Image of Texture

Table 3. Results of the Texture databases classification

MLP	Classification rate	99 %
	Iterations	8.75
RBF	Classification rate	**99.25 %**
	Iterations	25+5
NFC	Classification rate	99 %
	Iterations	1.5
LNFC	Classification rate	**99.25 %**
	Iterations	1.5

6 Conclusion

In this paper, a model of Neuro-Fuzzy Systems is presented and evaluated. This model is an improvement of the NFC. To evaluate the performances established by this development, the LNFC is compared with a conventional NFC and with two models of ANNs: MLP and RBF nets.

Three databases were used for the comparison with these classifiers. So, our model was examined by different type of feature: Length and width Measure (in Iris database); pixel value and their position (in the human thigh database) and calculation of different local correlations (in the texture database).

The obtained results on these databases show that the proposed approach improve performances of the NFC. In the other hand, the NFC is faster and more stable than

the MLP and RBFNN; the LNFC provides also this property because its conception does not require any modification in the structure and the training algorithm of the NFC.

The training of the LNFC is performed without modifying the membership functions parameters, which leads to obtaining simple training process and not loosing the original linguistic meaning of the membership functions. We can also change the T-norme operator (used in neurons of the third layer) and the membership functions without modifying the training algorithm.

References

1. Nemissi, M., Seridi, H., Akdag, H.: Labeled Neuro-Fuzzy Classification. Asian Journal of Information Technology 4 (2005) 868–872.
2. Nemissi, M., Boudouda, H., Seridi, H., Akdag H.: Classification Etiquetée Neuro Floue. 2ème Colloque International sur l'optimisation et les Systèmes d'information (COSI'05), Bejaia, Algérie, 12–14 juin (2005) 252–262.
3. Looney, C.: Pattern Recognition Using Neural Networks. Oxford University Press, New York (1997).
4. Zweiri, Y. H.: Whidborne, J. F., Seneviratne, L. D.: Three-term backpropagation algorithm. Neurocomputing 50 (2003) 305–318.
5. Zweiri, Y. H.: Optimization of a Three-Term Backpropagation Algorithm Used for Neural Network Learning. International journal of computational intelligence 3(2006) 322–327.
6. Roger Jang, J. S., Sun, C. T.:A neuro-fuzzy classifier and its applications. In proc. of IEE international conference on fuzzy systems, San Francisco 1 (1993) 94–98.
7. Chakraborty, D., Pal, N. R.: A Neuro-Fuzzy Scheme for simultaneous Feature Selection and Fuzzy Rule Based Classification, IEEE tans., Neural Networks 15-1 (2004) 110–123.
8. Masulli F., Sperduti A., Learning Techniques for Supervised Fuzzy Classifiers, Chapter 4, Fuzzy Learning and Applications, CRC press (2001) 147–169.
9. Lotfi A., Learning Fuzzy Systems, Chapter 6, Fuzzy Learning and Applications, CRC press (2001) 205–222.
10. Nemissi, M., Boudouda, H., Seridi, H.: Comparing performances of the MPL, RVFLNN and NFC for human thigh classification. Proc. of the International Workshop on Text, Image & Speech Recognition, Annaba- Algeria (2005) 125–131.

WordNet Based Multilingual Text Categorization

Bentaallah Mohamed Amine and Malki Mimoun

EEDIS Laboratory, Department of computer science
Sidi Bel Abbes, 22000. ALGERIA
mabentaallah@univ-sba.dz
malki-m@yahoo.com
http://www.univ-sba.dz

Abstract. This article is essentially dedicated to the problem of Multilingual Text Categorization, that consists in classifying documents in different languages according to the same classification tree. The proposed approach is based on the idea to spread the utilization of WordNet in Text Categorization towards Multilingual Text Categorization. Experimental results of the bi-lingual classification of the ILO corpus (with the documents in English and Spanish) show that the idea we describe are promising and deserve further investigation.

Key words: Multilingual, Text Categorization, WordNet, ILO corpus, Reuters-21578

1 Introduction

Text categorization is the task of assigning a Boolean value to each pair $< d_j, c_i > \in \mathcal{D} \times \mathcal{C}$, where $\mathcal{D}$ is the domain of documents and $\mathcal{C} = \{c_1, ..., c_{|c|}\}$ is a set of predefined categories. A value of T(True) assigned to $< d_j, c_i >$ indicates a decision to file d_j under c_i, while a value of F(False) indicates a decision not to file d_j under c_i [1].

During this last decade, research paid an important attention for the treatment of multilingual data due to the following several grounds:

- The availability of document collections distributed to the worldwide level created new needs to find information, whatever is the language and the storage support [2].
- The domination of the English language on the worldwide network is moving back to open the way toward a multilingual worldwide network [3]. Statistics showed that between 1998 and 2002, the population of which the English is the maternal language moved back from 60 to 36.5% [1].
- The time of globalization is coming. Many countries have been unified. The European project to unify European countries is a very important example

[1] These statistics are extracted from this website: http://www.nua.com/surveys/

in order to eliminate broader for the cooperation, global and large market, real international and free business. The high-developed technologies in network infrastructure and Internet set the platform of the cooperation and globalization. Indeed, the business should be global and worldwide oriented. Thus, the issues of the multilinguality arise and should be addressed as soon as possible in order to overcome the remaining technical barriers that still separate countries and cultures.

In this article, we propose a new approach for Multilingual Text Categorization that consists in spreading the use of WordNet in text categorization to categorize documents coming from different languages. The proposed approach is based on the translation of documents to be categorized towards the language of Shakespeare in order to be able to use the WordNet ontology thereafter. This hybridization between using machine translation and WordNet offers the following advantages:

- Without using machine translation, it becomes necessary to construct a WordNet ontology for each language. This construction is very expensive in terms of time and personals.
- The use of an ontology well constructed and rich as WordNet is going to permit to correct mistakes of the translation while using hypernymy relation and others.

The paper is organized as follows. In section 2, a definition of Multilingual Text Categorization is presented. In Section 3, we briefly review some related works for Multilingual Text Categorization. Section 4 is dedicated to presenting WordNet. We describe the proposed approach with all its stages in section 5. In section 6, we will evaluate our approach on two different datasets. Finally, conclusion and future works are reported in section 7.

2 Multilingual Text Categorization

Multilingual Text Categorization is a new area in text categorization in which we have to cope with two or more languages(e.g English, Spanish and Italian). In Multilingual Text Categorization, three scenarios can be distinguished:

- **Poly-lingual training:** In this scenario, the system is trained using training examples from all the different languages. A single classifier is built using a set of labelled train documents in all languages, which will classify documents from different languages. This scenario exclude the use of translation strategies, therefore, no distortion of information nor loss is not committed.
- **Cross-lingual training:** The system uses labelled training for only one language to classify documents in other languages. This approach is what we are interested in this paper. To solve this problem, we can use the translation in different ways:

  - **Training-Set Translation:** In this approach, the labelled set is translated into the target language which then is used to train a classifier for this language. So, the Cross-lingual training became a Poly-lingual training.
  - **Test-Set Translation:** This approach consists on translating the unlabelled documents into one language (L_1). To classify the unlabelled translated documents, the system is trained using the labelled training set for language (L_1). So, the Multilingual Text Categorization became Monolingual.
- **Esperanto language:** This approach uses an universal reference language which all documents are translated to. This universal language should contain all properties of the languages of interest and be organized in a semantic way.

3 Related Work

When we embarked on this line of research, we have note a lack on works addressing directly the area of Multilingual Text Categorization. The majority of research works essentially comes of the Multilingual Text Retrieval. Indeed, the two areas are based on the same aspects (similarity between texts, comparison of documents with queries or class profiles).

R.Jalam et al. presented in [4] three approaches for Multilingual Text Categorization that are based on the translation of documents towards a language of reference. The authors claimed to have got good enough results.

A.Gliozzo and C.Strapparava propose in [5] a new approach to solve the Multilingual Text Categorization problem based on acquiring Multilingual Domain Models from comparable corpora to define a generalized similarity function (i.e. a kernel function) among documents in different languages, which is used inside a Support Vector Machines classification framework. The results show that the approach largely outperforms a baseline.

These last years, researches showed that using ontologies in monolingual text categorization is a promising track. J.Guyot proposed in [6] a new approach that consists in using a multilingual ontology for Information Retrieval, without using any translation. He tried only to prove the feasibility of the approach. Nevertheless, it still has some limits because the used ontology is incomplete and dirty.

An intelligent methods for enabling concept-based hierarchical Multilingual Text Categorization using neural networks are proposed in [7]. These methods are based on encapsulating the semantic knowledge of the relationship between all multilingual terms and concepts in a universal concept space and on using a hierarchical clustering algorithm to generate a set of concept-based multilingual document categories, which acts as the hierarchical backbone of a browseable multilingual document directory. The concept-based multilingual text classifier is developed using a three-layer feed-forward neural network to facilitate the concept-based Multilingual Text Categorization.

4 WordNet & Text Categorization

WordNet [8] is a lexical inheritance ontology gifted with many different pointers that aim to represent some aspects of the semantics of the lexicon, and the relationships of different lexicalized concepts. Princetons WordNet has been under construction for over a decade and several versions were proposed. The last version (WordNet 2.1) contains more than 155000 word forms organized in 117597 word meanings. The word forms in WordNet are divided by part of speech into nouns, verbs, adjectives, and adverbs. The nouns are organized as a hierarchy of nodes, where each node is a word meaning or, as it is termed in WordNet, a synset. A synset is simply a set of words that express the same meaning in at least one context. For example,{accession, addition } is a synset which express the meaning of adding to something. Table 1 shows the distribution of the synsets on the four data bases (noun, verb, adjective, adverb) in WordNet 2.1 [2].

Table 1. Number of words&synsets in WordNet 2.1

POS	Word forms	Synsets
Noun	117097	81426
Verb	11488	13650
Adjective	22141	18877
Adverb	4601	3644
Totals	155327	117597

Synsets are connected to each other through various semantic relations. The most important relations between nouns are the relations of hyponymy and hypernymy, which are transitive relations between synsets. The hypernymy relationship between synsets A and B means that B is a kind of A. Hypernymy and hyponymy are inverse relationships, so if A is a hypernym of B, then B is a hyponym of A. For example the synset {computer, computing machine, computing device, data processor, electronic computer, information processing system} is a hypernym of the synset {home computer}. Usually each synset has only one hypernym, therefore this relation organizes WordNet into a hierarchical structure. Another pair of inverse relations that hold between nouns are the meronymy and the holonymy relations. If A is a holonym of B (or in other words B is a meronym of A), it means that B is a part of A. For example, synset {keyboard} is a meronym of the synset {computer, computing machine, computing device, data processor, electronic computer, information processing system}.

The wide coverage of WordNet and its free availability has promoted its utilization for a variety of text classification tasks, including IR and TC. While WordNet usage for text classification has not proven widely effective [9,10], some

[2] These statistics are extracted from this web site: http://wordnet.princeton.edu/

works in which WordNet synsets are used as indexing terms for IR and TC are very promising [11,12,13,14].

5 Our Approach

The suggested approach is composed of two phases as indicated in Fig. 1. The first relates on the training phase that consists on creating conceptual categories profiles. These conceptual profiles will contain concepts that characterize best a category with regard to the other categories. The second phase is the classification, it consists on using machine translation techniques to translate documents into English language in order to generate its conceptual vector using WordNet. After this, we have to weight document vector and categories profiles, then to calculate distances between them in order to be able to decide the adherence of the document to a category or an another one.

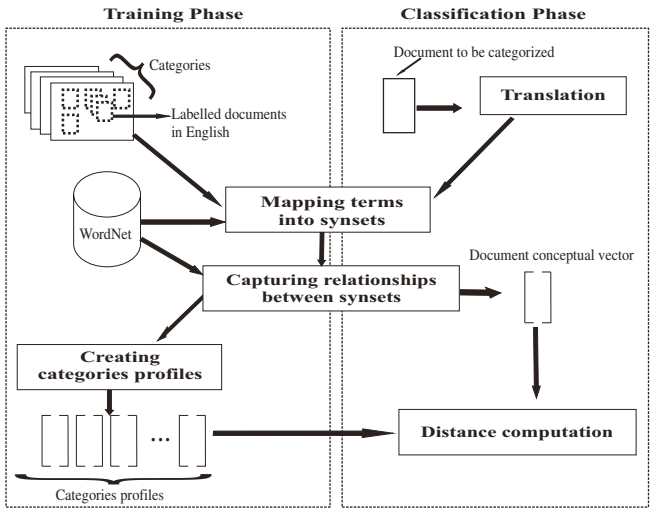


Fig. 1. Our approach

5.1 Training phase

The first issue that needs to be addressed in this phase is: how to represent texts so as to facilitate machine manipulation but also to retain as much information as needed ?. The commonly used text representation is the Bag-Of-Words, which simply uses a set of words with their number of occurrences to represent documents and categories [15]. This representation was disadvantaged by the ignorance of any relation between words, thus learning algorithms are restricted

to detect patterns in the used terminology only, while conceptual patterns remain ignored. In our approach, the training phase consists on using WordNet to create profiles categories which will contain concepts (synsets in WordNet) that characterize best one category with regard to the other categories. For this purpose, three steps are required:

- Mapping terms into synsets.
- Capturing relationships between synsets .
- Creating conceptual profiles categories.

Mapping terms into synsets The most straightforward representation of documents relies on term vectors. The major drawback of this basic approach for document representation is the size of the feature vectors, usually more than 10,000 terms. In the application of text categorization, however, completely different terms may represent the same concepts. In some cases, terms with different concepts can even be replaced with only one higher level concept without negative effect on performance of the classifier. Obviously, mapping terms to concepts is an effective and reasonable method to reduce the dimensionality of the vector space. In the most case, one word may have several meanings and thus one word may be mapped into several synsets which may add noise to the representation and may induce a loss of information. In this case, we need to determine which meaning is being used, which is the problem of sense disambiguation [16]. While there is a whole field of research dedicated to word sense disambiguation, it has not been our intention to determine which one could be the most appropriate, but simply whether word sense disambiguation is needed at all. For this purpose, WordNet returns an ordered list of sysnets for each term. Thereby, the ordering is supposed to reflect how common it is that a term reflects a synsets in "standard" English language. More common term meanings are listed before less common ones. In our approach, we used a simple disambiguation strategy that consists on considering only the most common meaning of the term (first ranked element) as the most appropriate. So our mapping process consists on replacing each term by its most common meaning. Thus the synset frequency is calculated by $sf(c_i, s) = tf(c_i, \{t \in T \mid first(Ref_s(t) = s\})$ where:

- c_i: the i^{th} category.
- $tf(c_i, T')$: the sum of the frequencies of all terms $t \in T'$ in the train documents of category c_i.
- $Ref_s(t)$: the set of all synsets assigned to term t in the WordNet.

Capturing Relationships between Synsets After mapping terms into synsets, this step consists on using the WordNet hierarchies to capture some useful relationships between synsets (hypernymy in our case). The synset frequencies will be updated as indicated in the following equation:

$$sf(c_i, s) = \sum_{b \in H(s)} sf(c_i, b) \quad (1)$$

Where:

- c_i: the i^{th} category.
- b and c are synsets.
- $H(s)$ contains the synsets that have the synset s as hypernym.

Creating conceptual categories profiles Selection methods for dimensionality reduction take as input a set of features and output a subset of these features, which are relevant for discriminating among categories [17]. Controlling the dimensionality of the vector space is essential for two reasons. The complexity of many learning algorithms depends crucially not only on the number of training examples but also on the number of features. Thus, reducing the number of features may be necessary to make these algorithms tractable. Also, although more features can be assumed to carry more information and should, thus, lead to more accurate classifiers, a larger number of features with possibly many of them being irrelevant may actually hinder a learning algorithm constructing a classifier. For our approach, a feature selection technique is necessary in order to reduce the big dimensionality by creating the conceptual categories profiles. For this purpose we used the χ_2 multivariate statistic for feature selection.

The χ_2 multivariate, noted $\chi_2^{multivariate}$ is a supervised method allowing the selection of features by taking into account not only their frequencies in each category but also interaction of features between them and interactions between features and categories. In our case, it consists on extracting, for each category, the K better synsets (our features) characterizing best the category compared to the others. With this intention, the matrix (synsets-categories) representing the total number of occurrences of the p synsets in the m categories is calculated. The total sum of the occurrences is noted N. The values N_{jk} represent the frequency of the synset s_j in the category c_k. Then, contributions of these synsets in discriminating categories are calculated as indicated in equation(2), then sorted by descending order for each category.

$$\mathcal{C}_{jk}^{\chi 2} = N \frac{(f_{jk} - f_{j.} f_{.k})^2}{f_{j.} f_{.k}} \times sign(f_{jk} - f_{j.} f_{.k}) \tag{2}$$

Where: $f_{jk} = \frac{N_{jk}}{N}$ representing the relative frequencies of the occurrences.

A positive $C_{jk}^{\chi 2}$ value indicates that it is the presence of the synset s_j which contribute in the discrimination of the category c_k while a negative value reveals that it is its absence which contribute in it.

5.2 Classification Phase

The classification phase consists on using the conceptual categories profiles in classifying unlabelled documents in different languages. Our classification phase consists on:

- Translating the document to be categorized and generating its conceptual vector;
- Weighting the conceptual categories profiles and conceptual vector of the unlabelled document;
- Calculating distance between the conceptual vector of the document and all conceptual categories profiles.

Translation and generation of the conceptual vector The translation of the text to be classified in the language of training corpus is also a paramount stage. The objective here is not to produce a translated text accurately recalling the semantic properties of the original text, but to provide a text ensuring a sufficient quality of classification. It is obvious that the result obtained will depend on the translator used. For that, we used JWT [3](Java Web Translator) library which provides automatic language translation for 14 languages including English, Spanish, French, Italian, Deutsch, Greek, Chinese, Japanese, Russian, ect. After translating document, we have to use WordNet in order to generate a conceptual vector for the document (Mapping terms into synsets and Capturing relationships between synsets).

Weighting This stage consists on weighting conceptual categories profiles and conceptual vector of the unlabelled document. Each weight $w(c, s)$ expresses the importance of synset s in vector of c with respect to its frequency in all training documents. The objective of using a feature weight rather than plain frequencies is to enhance classification effectiveness. In our experiments, we used the standard $tfidf$ function, defined as:

$$tfidf(s_k, c_i) = tf(s_k, c_i) \times \log(\frac{|C|}{df(s_k)}) \quad (3)$$

Where:

- $tf(s_k, c_i)$ denotes the number of times synset s_k occurs in category c_i.
- $df(s_k)$ denotes the number of categories in which synset s_k occurs.
- $\mid C \mid$ denotes the number of categories.

Distance computation The similarity measure is used to determine the degree of resemblance between two vectors. To achieve reasonable classification results, a similarity measure should generally respond with larger values to documents that belong to the same class and with smaller values otherwise. The dominant similarity measure in information retrieval and text classification is the cosine similarity between two vectors. Geometrically, it evaluates the cosine of the angle between two vectors $d1$ and $d2$ and is, thus, based on angular distance. This allows us to abstract from varying vector length. The cosine similarity can be calculated as the normalized dot product:

$$\mathcal{S}_{i,j} = \frac{\sum_{s \in i \frown j} tfidf(s,i) \times tfidf(s,j)}{\sqrt{\sum_{s \in i} tfidf^2(s,i) \times \sum_{s \in j} tfidf^2(s,j)}} \quad (4)$$

[3] This package is available on: http://sourceforge.net/projects/webtranslator

With:
s: a synset,
i and j: the two vector (profiles) to be compared.
$tfidf(s,i)$: the weight of the synset s in i.
$tfidf(s,j)$: the weight of the synset s in j.

Which can be translated in the following way: "If the two vectors are very alike their corresponding angle should be very small and approaching zero (cosine value approaching 1). On the other hand, if the angle is high, let say, 90 degrees, the vectors would be perpendicular (orthogonal) and the cosine value would be 0. In such case, the two vectors are not related". In our approach, this similarity measure is used to calculate distances between the conceptual vector of the unlabelled document and all categories profiles. As a result, the document will be assigned to the category whose profile is the closest with the document vector.

6 Experimental Results

6.1 Datasets for evaluation

Monolingual dataset The Reuters dataset has been used in many text categorization experiments; the data was collected by the Carnegie group from the Reuters newswires in 1987. There are now at least five versions of the Reuters datasets widely used in TC community. We choose the Modapte[4] version of the

Table 2. Details of the used Reuters-21578 categories

Class description	♯ Training	♯ Test
Earn	2877	1087
Acquisition	1650	719
Money-fx	538	179
Grain	433	149
Crude	389	189
Trade	369	118
Interest	347	131
Wheat	212	71
Ship	197	89
Corn	182	56
Total	7194	2788

Reuters-21578 collection of new stories. In our experiments, we used the 10 most frequent categories from this corpus as our monolingual dataset for training and testing (as shown in Table2).

[4] http://www.daviddlewis.com/ressources/testcollections/reuters21578

Multilingual dataset The ILO corpus is a collection of full-text documents, each labelled with one category (mono-classification) which can be downloaded from the ILOLEX website of the International Labour Organisation[5]. ILOLEX describes itself as a trilingual database containing ILO Conventions and Recommendations, ratification information, comments of the Committee of Experts and the Committee on Freedom of Association, representations, complaints, interpretations, General Surveys, and numerous related documents. The languages concerned are English, Spanish and French. In our experiments, we used a bilingual version (with documents in English and Spanish) mono-classified in 10 categories with a rather varying number of documents per category as shown in Table 3.

Table 3. The 10 used Categories of the ILO corpus

Category	♯ English	♯ spanish
Special prov. by Sector of Econ. Act.	108	121
Conditions of employment	397	86
Conditions of work	299	71
Economic and social development	22	23
Employment	410	448
Labour Relations	276	278
Labour Administration	85	81
Health and Labour	98	86
Social Security	150	148
Training	79	20
Total	1924	1362

6.2 Evaluation method

Experimental results reported in this section are based on the so-called "F_1 measure", which is the harmonic mean of precision and recall.

$$F_1(i) = \frac{2 \times precision \times recall}{precision + recall} \tag{5}$$

In the above formula, precision and recall are two standard measures widely used in text categorization literature to evaluate the algorithms effectiveness on a given category. We also use the macroaveraged F_1 to evaluate the overall performance of our approach on given datasets. The macroaveraged F_1 computes the F_1 values for each category and then takes the average over the per-category F_1 scores. Given a training dataset with m categories, assuming the F_1 value for the $i-th$ category is $F_1(i)$, the macroaveraged F_1 is defined as :

$$MacroAveragedF_1 = \frac{\sum_{i=1}^{m} F_1(i)}{m} \tag{6}$$

[5] http://ilolex.ilo.ch:1567/Spanish/index.htm

6.3 Results

In order to be able to show the utility of the use of WordNet in multilingual categorization, we tested the suggested approach on both Monolingual and Multilingual datasets. It is necessary to specify here, that our objective is not to compare the two used datasets but to show if it is possible to spread the use of WordNet in Multilingual Text Categorization with using machine translation techniques. The results of the experimentations are presented in Table 4.

Table 4. Comparison of MacroAveraged F_1 results on the two used datasets

Size of profiles	Monolingual dataset	Multilingual dataset
k=100	0.664	0.416
k=200	0.663	0.455
k=300	0.663	0.504
k=400	0.666	0.534
k=500	0.667	0.554
k=600	0.668	0.560
k=700	0.667	0.561
k=800	0.667	0.570
k=900	0.668	0.573

Concerning the profiles size, it is noted that for the two datasets, the best performances are obtained with size profile $k = 900$. Indeed, the performances improve more and more by increasing the size of profiles.

In addition, it is noticed that the performances of multilingual categorization are very close to those of monolingual categorization. Indeed, the differences of the error rates obtained in multilingual categorization (after translation) compared with those obtained in monolingual categorization are not significant.

7 Conclusion

In this paper, we proposed a new approach for Multilingual Text Categorization which is based on the one hand on the use of WordNet and other hand on the use of machine translation techniques. The obtained results are encouraging and carry out us to confirm that the use of WordNet in Multilingual Text Categorization is a promising track.

Our future works will concern the use of WordNet in calculation of distance in order to be able to test the use of the semantic distances in stead of the statistics distances. Another track consists on using other disambiguation, selection and Weighting techniques.

References

1. Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys. (2002) 1–47.
2. Peters, C., Sheridan, P.: Accès multilingue aux systèmes d'information. In 67th IFLA Council and General Conference. (2001).
3. Nunberg, G.: Will the Internet speak english?. The American Prospect. (2000).
4. Jalam, R., Clesh, J., Rakotomalala, R.: Cadre pour la catégorisation de textes multilingues. 7 èmes Journées internationales d'Analyse statistique des Données Textuelles. Louvain-la-Neuve, Belgique (2004) 650–660.
5. Gliozzo, A.M., Strapparava, C.: Cross Language Text Categorization by acquiring Multilingual Domain Models from Comparable Corpora. In Proceedings of the ACL Workshop on Building and Using Parallel Texts. Ann Arbor, Michigan, USA (2005) 9–16.
6. Guyot, J., Radhouani, S., Falquet, G.: Ontology based multilingual information retrieval. In CLEF Workhop, Working Notes Multilingual Track, Vienna, Austria (2005) 21–23.
7. Chau, R., Yeh, C.H, Smith, K.:A Neural Network Model for Hierarchical Multilingual Text Categorization. In proceeding of ISSN-05 Second International Symposium on Neural Networks, Chongqing, China (2005) 238–245.
8. Miller, G.A.:WordNet: An On-Line Lexical Database. In Special Issue of International Journal of Lexicography, Vol 3, No. 4 (1990).
9. Scott, S.: Feature engineering for a symbolic approach to text classification. Master's thesis, Computer Science Dept., University of Ottawa, Ottawa, CA (1998).
10. Voorhees, E.: Using WordNet for text retrieval. In WordNet: An Electronic Lexical Database. MIT Press (1998).
11. Fukumoto, F., Suzuki, Y.: Learning lexical representation for text categorization. In Proceedings of the NAACL 2001 Workshop on WordNet and Other Lexical Resources. (2001).
12. Gonzalo, J., Verdejo, F., Chugur, I., Cigarran, J.: Indexing with WordNet synsets can improve text retrieval. In Proceedings of the COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems. (1998).
13. Petridis, V., Kaburlasos, V., Fragkou, P., Kehagias, A.: Text classification using the FLNMAP neural network. In Proceedings of the 2001 International Joint Conference on Neural Networks. (2001).
14. Mihalcea, R., Moldovan, D.: Semantic indexing using WordNet senses. In Proceedings of ACL Workshop on IR and NLP. (2000).
15. Peng, X., Choi, B.: Document Classifications Based On Word Semantic Hierarchies. In proceedings of the IASTED International Conference on Artificial Intelligence and Applications. Innsbruck, Austria (2005) 362–367.
16. Ide, N., Vieonis, J.: Introduction to the special issue on word sense disambiguation: The state of the art. Computational Linguistics. (1998).
17. Dash, M., Liu, H.: Feature selection for classification. Intelligent Data Analysis. (1997).

Algorithmes de sélection dans un problème d'ordonnancement en présence des demandes stochastiques

Djamel Berkoune [1], Khaled Mesghouni [2]

[1] *GIPSA–Lab INPG / ENSIEG BP*46, *Rue de la Houille Blanche* 38402 *St Martin d'Hères - Cedex FRANCE*
[2] *LAGIS – UMR CNRS* 8146, *Ecole Centrale de Lille, BP* 48, 59651 *Villeneuve d'Ascq Cedex, France*
Djamel.Berkoune@gmail.com, Khaled.Mesghouni@ec-lille.fr

Résumé : Dans ce papier, nous traitons l'un des problèmes survenant dans les ateliers de production à savoir la prise en compte dans la solution d'ordonnancement proposée la production de produits prévisionnels. La liste des demandes est divisée en demandes fermes et demandes prévisionnelles. On souhaite minimiser la durée totale de production ″Makespan″. Pour cela, on utilise les méthodes métaheuristiques comme les algorithmes génétiques pour l'optimisation des demandes fermes et les demandes prévisionnelles seront insérés, selon des heuristiques conçues spécialement, dans les solutions réelles données par les algorithmes génétiques. L'objectif principal est de sélectionner la solution des demandes fermes qui sera la meilleure solution après insertion des demandes prévisionnelles, minimisant ainsi le Makespan de l'ensemble des demandes (fermes et prévisionnelles).

Mots-clés : *Ordonnancement job shop, algorithmes génétiques, Makespan, Coût de production, Demandes fermes et demandes prévisionnelles.*

1. Introduction

L'ordonnancement a pour but d'organiser la production à court terme. Il planifie des tâches sur des ressources pour assurer la fabrication et la livraison des commandes en temps et en heure. Dans cette planification, la date de livraison est en général une donnée du problème. Pourtant, lorsque les demandes sont en cours de négociation (prévisionnelles), cette date n'est pas encore déterminée et il est alors intéressant de pouvoir faire un ordonnancement prévisionnel, intégrant ces demandes encore incertaines, à partir duquel on peut évaluer le délai de fabrication des produits en fonction de la charge de l'atelier. La principale difficulté dans la réalisation de cet ordonnancement et l'évaluation du délai de fabrication, avec la prise en compte de toutes les demandes susceptibles d'occuper l'atelier, alors qu'une partie d'entre elles est incertaines. En pratique, les problèmes d'ordonnancement nécessitent souvent la prise en compte de plusieurs critères, tel que les problèmes survenant dans les ateliers de production pour lesquels on souhaite généralement maximiser la quantité de produits réalisés tout en satisfaisant au mieux les délais de livraison et en minimisant les coûts de production. Ce type de problème est hautement combinatoire et appartient souvent à la classe NP-difficile, sa résolution nécessite l'application de méthodes et algorithmes spécifiques. Notre problème consiste à proposer un ordonnancement optimal au sens de plusieurs critères, pour cela, il existe plusieurs méthodes qui transforment ces problèmes multicritères en problèmes unicritère. Le but

principal est de générer une variété de solution Pareto optimales diversifiées dans l'espace de recherche.

Dans notre travail on s'intéresse aux problèmes d'ordonnancement avec le modèle de demande incertaine dont le but est de minimiser la durée totale de production (*Cmax*) et/ou le coût de production. La liste des demandes est divisée en deux sous ensembles : demandes fermes qui seront ordonnancées en priorités et les demandes prévisionnelles (demandes en cours de négociation, tâches de maintenance,…) qui seront insérées dans les disponibilités des machines. L'objectif est de développer une méthode de sélection qui nous permettra de choisir parmi les solutions proposées des demandes fermes, celle qui sera la meilleure après l'opération d'insertion des demandes prévisionnelles dans les disponibilités machines restantes. Concrètement, le problème se divise en deux parties ; La première concerne l'ordonnancement des demandes fermes avec les algorithmes génétiques qui sont bien adaptés pour ce type de problème [7]. Le taux de convergence et la qualité de la solution sont améliorés avec un codage spécifiquement conçu et des opérateurs génétiques adaptés pour des problèmes d'ordonnancement [3, 5, 6, 7]. La deuxième partie se consacre à l'insertion des demandes prévisionnelles dans les solutions réelles (ordonnancement des demandes fermes) selon les méthodes d'insertion présentées dans [1] qui permettent d'insérer ces demandes dans les disponibilités machines en minimisant les critères considérés (*Cmax* et *coût de production*). L'approche présentée nous permet de proposer au décideur, la meilleure solution des demandes fermes, ainsi que, la meilleure solution trouvée en incluant les demandes prévisionnelles (fermes+ prévisionnelle). Cette démarche quantifie ainsi en termes de coût et de temps la décision de fabriquer ou pas des demandes prévisionnelles.

Cet article est organisé comme suit : dans la section 2 nous définissons notre problème et les notations utilisées. La section 3 contient une description de la méthode de résolution utilisée. Enfin, la dernière section contient quelques exemples d'illustration.

2. Description du problème d'ordonnancement

Le problème consiste à organiser l'exécution d'un ensemble de N jobs sur un ensemble de M machines. Chaque job j contient un certain nombre n_j d'opérations. Chaque opération i du job j notée O_{ij} doit être exécutée sur une des machines avec une date de début au plus tôt r_{ij} (la date de début au plus tôt du job j est $r_j = r_{1j}$) et une contrainte de précédence ($r_{i+1,j} \geq r_{ij} + p_{ijm}$). A chaque opération O_{ij} est associée un ensemble de durées opératoires sur l'ensemble des machines (p_{ijm} ; $1 \leq m \leq M$). Un ordonnancement consiste alors à attribuer une machine m qui sera affectée à l'exécution de O_{ij} ($x_{ijm} = 1$, avec x_{ijm} le coefficient d'affectation de l'opération O_{ij} sur la machine m); et une date de début pour chaque O_{ij}. La date de fin du job est notée C_j et la date de fin de l'opération O_{ij} sur la machine k est C_{ijk}. L'objectif est de trouver la date de début d'exécution pour chaque opération O_{ij} et la machine affectée en respectant toutes les contraintes considérées, ainsi que d'ordonnancer les jobs de façon à minimiser les critères considérés.
Les hypothèses de ce problème sont :

- Les machines sont indépendantes les unes des autres ;
- Toutes les machines sont disponibles à la date $t = 0$;
- Tous les jobs peuvent être commencés à la date $t = 0$;
- Un ensemble de jobs dont n fermes, c'est à dire des commandes fermes qui sont prioritaires, N-n prévisionnels (commandes incertaines);
- Une machine ne peut exécuter qu'une seule opération à un instant donné (contrainte de disponibilité) ;
- Une opération en cours d'exécution ne peut être interrompue (interdiction de préemption) ;
- Contrainte de précédence des opérations, $r_{i+1,j} \geq r_{i,j} + p_{i,j,k}$;
- Les jobs sont indépendants les uns des autres.

Les objectifs considérés sont :

- minimiser le Makespan noté ***Cmax :***

$$Cmax = max \{C_j \mid j = 1,..., N\} \quad (1)$$

Où C_j est la date de fin du job *j*

- minimiser le coût de production noté **C** :

$$C = \sum_{j=1}^{N} F_j + \sum_{m=1}^{M} L_m + \sum_{j=1}^{N} S_j + \sum_{j=1}^{N} P_j \quad (2)$$

Où L_m est le coût de lancement de la machine *m,* S_j est le coût de stockage de job *j*, et P_j est le coût de pénalité.

- *Le coût de fabrication du job j (**F_j**)* :

$$F_j = MP_j + \sum_{m=1}^{M} \sum_{j=1}^{n_j} x_{ijm} CMmp_{ijm} \quad (3)$$

Où MP_j est le coût de la matière première du job *j* et CM_m est le coût unitaire sur la machine *m*

- *Le coût de stockage du job prévisionnel j sera en fonction de la probabilité de stockage Pr[d] (**S_j**)* :

$$S_j = \sum_{d \in D_j} Cs_j \cdot E_j(d) \cdot \Pr[d] \text{ ; Avec } E_j(d) = |\min \{0, C_j - d_j\}| \quad (4)$$

Où Cs_j est le coût de stockage du job *j*, E_j est la durée de stockage et d_j est date de livraison du job *j*.

3. Approche de résolution proposée

Notre approche consiste à ordonnancer la totalité des jobs en deux étapes. La première étape consiste à ordonnancer les jobs fermes avec les algorithmes génétiques indiqués dans [1]. La deuxième étape consistera à insérer les jobs prévisionnels dans le plan de production des machines en utilisant la méthode d'insertion présentée dans [1]. Cette méthode permet l'insertion des demandes prévisionnelles dans les disponibilités machines sans modifier la solution initiale. Nous décrivons ces méthodes dans les sections suivantes.

3.1 Les Algorithmes Génétiques

La première description rigoureuse du processus des algorithmes génétiques (AGs) a été donnée par Holland en 1960 [6]. Le but de ces AGs est d'optimiser une fonction prédéfinie, fonction coût appelée *"fitness"*, ils travaillent sur un ensemble de solutions appelé *"population"* d'individus. Ces derniers sont constitués d'un ensemble de *"gènes"* qui peuvent prendre plusieurs valeurs appelées *"allèles"* [8]. Une première population est choisie soit aléatoirement, soit par des heuristiques ou par des méthodes spécifiques au problème, soit encore par mélange de solutions aléatoires et heuristiques, cette population doit être suffisamment diversifiée pour que l'algorithme ne reste pas bloqué dans un optimum local. Le codage utilisé dans notre travail est présenté dans [7]. Dans ce codage un individu est représenté par une matrice, où les lignes représentent la gamme de chaque job. Chaque cellule de cette ligne (représentant une opération) contient deux termes. Le premier indique le numéro de la machine qui est affectée à cette opération, le second représente la date de début d'exécution de l'opération. Cette date est calculée en tenant compte des contraintes de ressource et de précédence [7]. D'une façon générale un chromosome se présente comme suit : $J_j : (m_k, t_m)$ m_k est la machine qui est affectée à l'exécution de l'opération, t_m est la date de début d'exécution sur la machine m_k. Le processus

d'amélioration des individus s'effectue par utilisation d'opérateurs génétiques qui sont : la sélection, le croisement et la mutation [5, 9]**.**

3.2 Procédure d'insertion

Insertion des jobs prévisionnels dans les périodes de disponibilité de la machine. Cette procédure nous permet d'insérer les jobs prévisionnels dans les périodes de disponibilité des machines sans modifier la solution initiale d'ordonnancement des jobs fermes. Il existe plusieurs heuristiques d'insertion des tâches dans un plan de production (SPT, LPT, EDD…). Dans notre problème nous avons utilisé l'heuristique *ECT* (*Earliest Completion Time*) qui considère en priorité l'opération qui possède le temps d'accomplissement le plus court, parce qu'elle vise intuitivement à réduire la date de fin donc le *Cmax*.

I_{ij} : ensemble d'insertion possible pour l'opération O_{ij}
n : nombre de jobs fermes
Pour ($j=n+1$;$j<=N$)
  Pour ($i=1$;$i<= n_j$) $I_{ij} = \varnothing$
    Pour ($k=1$;$k<=M$)
      Si durée recherchée ($p_{i,j,k}$) *correspond à la période de disponibilité de la machine k et toutes les contraintes sont vérifiées*
        alors :
          - Mémoriser cette possibilité d'insertion et le temps de fin de cette opération C_{ijk} *:* $I_{ij} = I_{ij} \cup \{(k, C_{ijk})\}$
    Fin pour
        - Insérer l'opération O_{ij} sur la machine k dans I_{ij} avec C_{ijk} est la plus petite date de fin de l'opération
  Fin Pour
Fin pour

3.3 Approche de sélection

Les travaux précédents [2] montrent que la meilleure solution des demandes fermes avant l'insertion des demandes prévisionnelles n'est pas nécessairement la meilleure solution après insertion de ces demandes. Notre objectif dans cet article est de sélectionner la solution des demandes fermes qui sera la meilleure après insertion des demandes prévisionnelles. Pour cela, nous proposons une nouvelle approche qui nous permet de sélectionner cette solution. La difficulté réside dans la mise au point d'un critère permettant de savoir si une bonne solution proposée pour les jobs fermes reste bonne après insertion des jobs prévisionnels. Pour cela, nous avons mis au point une méthode (un critère à calculer) basé sur une combinaison entre les durées de disponibilités des machines après ordonnancement des demandes fermes ($Dispo = C_{\max} * M - durée\ d'occupation\ des\ machines$) et la moyenne des durées opératoires des opérations des demandes prévisionnelles

$Moy = \dfrac{\left(\sum_{i=1}^{n_j}\sum_{j=n+1}^{N}\sum_{k=1}^{M} p_{ij'k}\right)}{M}$ donnant ainsi le critère *"select"* représentant le rapport entre les disponibilités des machines après ordonnancement des jobs fermes et la moyenne des durées opératoires des jobs prévisionnels :

$$Select = \frac{Dispo}{Moy} = \frac{C_{\max} * M - duration\ \ of\ \ occupation\ \ machines}{(\sum_{i=1}^{nj}\sum_{j=n+1}^{N}\sum_{k=1}^{M} p_{ij'k}) / M} \qquad (5)$$

Cette méthode a été testée sur plusieurs exemples, nous avons remarqué que 80% des résultats trouvés montrent que si un ordonnancement des demandes fermes a une valeur de *Select* ≥ 60% cette solution sera la meilleure après insertion des demandes prévisionnelles à condition qu'elle soit dans le premier rang des solutions Pareto

4. Exemple

Nous avons illustré cette méthode sur un exemple de 6 jobs N=6 (3 fermes n=3 et 3 prévisionnels n'=3) et 3 machines M=3. Chaque job possède 3 opérations n_j=3, les durées opératoires sont indiquées dans le tableau suivant :

		M1	M2	M3
J1	O_{11}	1	9	3
	O_{21}	3	5	2
	O_{31}	6	7	1
J2	O_{12}	1	4	5
	O_{22}	2	8	4
	O_{32}	9	5	1
J3	O_{13}	5	4	3
	O_{23}	1	4	1
	O_{33}	3	2	2
J4	O_{14}	1	2	3
	O_{24}	3	2	2
	O_{34}	3	4	2
J5	O_{15}	1	2	3
	O_{25}	2	2	3
	O_{35}	1	3	2
J6	O_{16}	1	2	3
	O_{26}	2	2	2
	O_{36}	1	3	2

Tableau I. *Durées d'exécution des opérations sur les machines.*

Les coûts liés aux jobs et aux machines sont indiqués dans le tableau suivant :

	J1	J2	J3	J4	J5	J6
Prix de vente	45€	45€	70€	40€	30€	45€
Matière 1ère	5€	4€	5€	5€	5€	5€

Tableau II. *Coûts lies à chaque job*

	Probabilités de stock des jobs prévisionnels								
Date de livraison d_j	9	10	11	12	13	14	15	16	17
Probabilité de livraison (%)	5	10	10	5	5	20	15	15	15

Tableau III. *Probabilités de livraison des jobs prévisionnels*

	M1	M2	M3
Coût de production (€/ut)	5	1	2
Coût de lancement (€)	5	1	2

Tableau IV. *Coûts liés à chaque machine*

Après minimisation du *Cmax* avec les algorithmes génétiques et après insertion des jobs prévisionnels on trouve les résultats indiqués dans la figure 1.

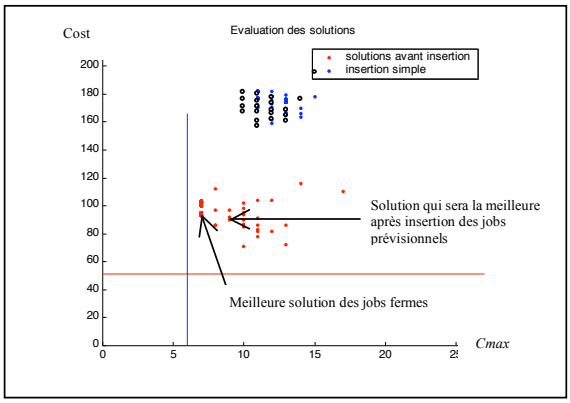


***Figure. 1**. Ensemble de solutions avant et après insertion des jobs prévisionnels.*

Dans la Figure 1 la meilleure solution avant insertion des jobs prévisionnels nous donne les résultats suivants :
Cmax=7, un coût de production de 93€, *Select*=31%, Moy≈19, Dispo=6, la solution est représentée dans la Figure 2.

date	1	2	3	4	5	6	7
M1	O_{11}	O_{12}	O_{22}	O_{22}			
M2	O_{13}	O_{13}	O_{13}	O_{13}		O_{33}	O_{33}
M3		O_{21}	O_{21}		O_{23}	O_{31}	O_{32}

Figue. 2. *Cmax=7ut; C=93€; Select =28%,*

Après insertion des jobs prévisionnels dans les disponibilités de cette solution on trouve un *Cmax*=13 et un coût de production de 174.40€. La solution est représentée dans la figure suivante :

date	1	2	3	4	5	6	7	8	9	10	11	12	13
M1	O_{11}	O_{12}	O_{22}	O_{22}	O_{15}	O_{25}	O_{25}	O_{35}	O_{16}			O_{36}	
M2	O_{13}	O_{13}	O_{13}	O_{13}		O_{33}	O_{33}	O_{14}	O_{14}	O_{24}	O_{24}		
M3		O_{21}	O_{21}		O_{23}	O_{31}	O_{32}			O_{26}	O_{26}	O_{34}	O_{34}

Figure. 3. *Cmax=13ut; C=174,40€;*

Par contre la meilleure solution après insertion des jobs prévisionnels nous donne un *Cmax*=11 et un coût de production de 170.50€, la représentation de cette solution est la suivante :

Date	1	2	3	4	5	6	7	8	9	10	11
M1	O_{11}	O_{12}	O_{22}	O_{22}	O_{15}	O_{14}	O_{16}	O_{35}	O_{26}	O_{26}	O_{36}
M2	O_{13}	O_{13}	O_{13}	O_{13}		O_{25}	O_{25}	O_{24}	O_{24}		
M3		O_{21}	O_{21}		O_{23}	O_{31}	O_{32}	O_{33}	O_{33}	O_{34}	O_{34}

Figure. 4. *Cmax=11ut; C=170,50€;*

La solution initiale (solution des demande fermes) qui nous a permet de trouver cette solution possède un *Cmax*=9, un coût de production de 90€, *Select* = 63,5%, Moy ≈19, *Dispo*=12. Figure.5 représente cette solution.

Date	1	2	3	4	5	6	7	8	9
M1	O_{11}	O_{12}	O_{22}	O_{22}					
M2	O_{13}	O_{13}	O_{13}	O_{13}					
M3		O_{21}	O_{21}		O_{23}	O_{31}	O_{32}	O_{33}	O_{33}

Figure. 5. *Cmax=9ut; C=90€; Select =70%,*

Avec notre approche de sélection on trouve directement cette solution qui sera la meilleure après insertion des jobs prévisionnels. Afin de montrer l'efficacité de notre approche, on a testé cette méthode sur plusieurs exemples, les données de ces exemples sont présentées dans la colonne *″Données″* dans le tableau V, ces données ont été générées aléatoirement : le nombre de jobs *N* varie entre 6 et 32, le nombre de machines *M* entre 6 et 15 et les durées de chaque opération sur l'ensemble de machines P_{ijk} varie entre 1 et 10.
Dans ce tableau les meilleures solutions des jobs fermes avant l'insertion des jobs prévisionnels sont représentées dans la colonne *″* $Cmax_{f1}$*″*, et les solutions trouvées après insertion selon la méthode présentée plus haut sont représentées dans la colonne *″* $Cmax_{p1}$*″*.
La meilleure solution des jobs prévisionnels est représentée dans la colonne *″* $Cmax_{p2}$*″* et la solution initiale dont on a inséré ces jobs prévisionnels est représentée dans la colonne *″* $Cmax_{f2}$*″*.

Données				meilleure solution pour les jobs fermes				meilleure solution pour les jobs prévisionnels			
n	*n'*	*M*	*Av*	$Cmax_{f1}$	*A*	*Select*	$Cmax_{p1}$	$Cmax_{f2}$	*A*	*Select*	$Cmax_{p2}$
12	16	13	50	**7**	29	**0.58**	**9**	**8**	33	**0.66**	**8**
3	3	9	47	**4**	29	**0.58**	**7**	**6**	45	**0.96**	**6**
4	5	7	65	**4**	20	**0.31**	**9**	**7**	38	**0.62**	**8**
10	4	10	68	**8**	24	**0.35**	**10**	**9**	42	**0.62**	**9**
9	9	6	75	12	51	0.68	12	12	51	0.68	12
7	5	9	83	**6**	42	**0.51**	**9**	**7**	50	**0.60**	**8**
5	7	10	77	**3**	24	**0.31**	**8**	**6**	51	**0.66**	**7**
12	13	15	50	13	49	0.98	14	13	49	0.98	14
13	19	8	51	10	35	0.69	12	10	35	0.69	12
20	9	10	65	**12**	32	0.49	14	13	51	**0.78**	**13**
25	4	13	63	15	42	0.66	17	15	42	0.66	17
22	6	6	70	**12**	26	0.37	19	14	59	**0.84**	**17**
27	3	9	82	**11**	35	0.42	14	12	63	**0.76**	**12**
21	6	8	76	**13**	28	0.36	15	14	57	**0.75**	**14**

Tableau V. Comparaison des résultats des jobs fermes et des jobs prévisionnels.

Avec la méthode proposée on arrive à sélectionner la meilleure solution en minimisant le *Cmax* de l'ensemble des jobs (fermes et prévisionnels). Alors que dans les travaux précédents [2] on satisfait en priorité la minimisation du *Cmax* des jobs fermes puis on insère les jobs prévisionnels ce qui ne garantie pas une meilleur solution au sens du ou des critères considérer pour l'ensemble du problème c'est-à-dire une solution optimale (proche de l'optimale) pour les jobs fermes et prévisionnels.

5. Conclusion

Les travaux précédents [2] ont montrés que la meilleure solution des jobs fermes avant insertion des jobs prévisionnels n'est pas forcément la meilleure solution après l'insertion. Dans ce travail, nous avons présenté une nouvelle approche de résolution de sélection avec comme objectif l'optimisation du Makespan (*Cmax*) de l'ensemble des jobs. Cette méthode montre comment choisir la solution qui sera la meilleure après insertion des jobs prévisionnels. Le travail actuel consiste à tester cette méthode sur des exemples de grande taille pour améliorer cette méthode. Nos travaux futurs consistent à intégrer d'autres critères tels que la charge des machines et de tenir compte des changements qui arrivent au moment de la production tel que les pannes des machines, …etc.

6. References

[1] D. Berkoune, K. Mesghouni, B. Rabenasolo, *Approches évolutionnistes pour le problème d'ordonnancement prévisionnel*. Conférence CIFA2004, 2004, Douz. Tunisie

[2] D. Berkoune, K. Mesghouni, B. Rabenasolo. *Optimisation mutli-objecifs avec des demandes prévisionnelles*. Journée Doctorales et Nationale de MACS JDMACS-JNMACS2005, 2005, Lyon. France.

[3] J. Carlier., P. Chrétienne., *Problème d'ordonnancement Modélisation/complexité Algorithmes*. Edition Masson, 1988.

[4] F. Glover, M. Laguna, *Tabu search*, Kluwer Academic Publishers, 1997

[5] D.E Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, 1989.

[6] J.H. Holland. *Adaptation in natural and artificial system*, University of Michigan Press 1975

[7] K. Mesghouni. *Application des algorithmes évolutionnistes dans les problèmes d'optimisation en ordonnancement de la production*. Thèse université de Lille 1, 1999

[8] J.M. Renders. *Algorithmes génétiques et réseaux de neurones*, Edition Hermès1995.

[9] G.Syswerda., *Schedule Optimization Using Genetic Algorithm*. In Handbook of Genetic Algorithm, Van Nostrand Reinhold, New York, 1990.

Décodage acoustique phonétique de la langue arabe : Utilisation des réseaux gamma pour la reconnaissance des consonnes emphatiques

Abderrahmane Bendahmane[1], Abderrahmane Louni[2], Abdelkader Benyettou[3]

Laboratoire SIgnal-IMage-PArole(SIMPA), Dépt. Informatique
Faculté des Sciences, Université des Sciences et de la Technologie d'Oran – USTO, Algérie. BP. 1505, Oran El-Mnaouer 31000.

[1]abder.bendahmane@gmail.com
[2]louniab@gmail.com
[3]aek.benyettou@univ-usto.dz

Résumé. La langue arabe connaît une série de consonnes complexes dites « emphatiques ». Leur reconnaissance consiste en l'occurrence un réel défie pour tout système de Décodage Acoustico Phonétique (DAP). Le but de notre travail est d'implémenter un module de DAP capable de reconnaître ces consonnes, caractéristiques de la langue arabe. Notre problème nous pousse à utiliser un système capable de modéliser à la fois des connaissances phonétiques instantanées, reposant sur des vecteurs acoustiques, et des connaissances phonétiques temporelles, reposant sur la durée antérieure de certains phénomènes. Nous utiliserons pour cela une architecture neuro-temporelle assez récente qui est le réseau gamma. Grâce à l'implantation de filtres gamma au niveau des récurrences de ces neurones, ce réseau constitue une véritable mémoire dynamique temporelle. Cette caractéristique est idéale pour la classification phonétique et le place parmi les meilleurs classificateurs temporels actuels.

Mots clés: Burst, DAP, Emphatique, Formant, Langue arabe, MFCC, MLP, RAP, Réseau gamma, TDNN.

1 Introduction

La simplicité de la parole en fait le moyen de communication le plus populaire entre êtres humains. Néanmoins, cette simplicité (pour l'être humain) renferme un traitement complexe fait par notre cerveau, de la production de la parole jusqu'à sa perception et sa compréhension, ce qui rend la parole difficilement automatisable pour une machine. La reconnaissance automatique de la parole continue (RAP) passe par la segmentation du signal de la parole en unités minimales (syllabe, phonème etc…) puis par l'identification de ces unités. Cette approche est très bien adaptée pour les systèmes à grand vocabulaire et pour la parole continue. Le DAP est fondamental dans une telle approche. C'est un module qui utilise essentiellement les

caractéristiques acoustiques pour obtenir, à partir du signal un ensemble de segments et dont extraire un vecteur de traits acoustiques (pertinents, discriminants, robustes) dont il tire une transcription phonétique (suite de phonèmes).

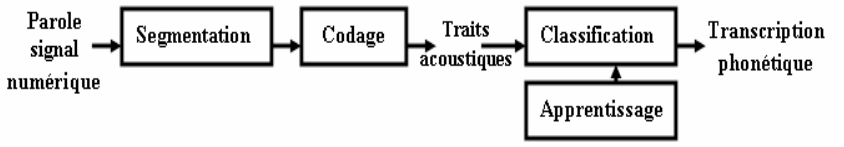


Fig. 1. Schéma général d'un module de DAP

La langue arabe connaît une série de consonnes complexes dites « emphatiques ». Il s'agit d'un trait articulatoire spécifique qui est la « pharyngalisation », le son produit est plus grave que pour le son non emphatique correspondant. Il est très important de bien distinguer le son emphatique du non emphatique correspondant car sa présence change complètement la signification d'un mot, par exemple :

سيف « épée » sœyf **صيف** « été » sayf

L'orthographe de l'arabe constitue en l'occurrence une aide. Souvent la présence d'une consonne emphatique dans un mot "contamine" l'environnement consonantique et vocalique, et c'est toute la syllabe qui est emphatisée. Dans certains mots, le ل et le ر sont emphatisés (en arabe dialectal et aussi en arabe standard). Il faut mémoriser ces mots car ces réalisations emphatiques ne sont pas marquées à l'écrit [7].

Table 1. Les consonnes emphatiques de la langue arabe et leurs variantes non emphatiques.

Non Emphatique	س	د	ذ	ت	ل	ر
Emphatique	ص	ض	ظ	ط	ل	ر

Le but de notre travail est de construire un système DAP capable de reconnaître des consonnes emphatiques des non emphatiques. Notre approche est basée sur un corpus d'apprentissage composé de fichiers au format «wav» soigneusement étiquetés. Pour la phase de segmentation nous nous contenterons d'un simple fenêtrage de taille fixe du signal de parole. Pour la phase de codage nous utiliserons des coefficients MFCC très réputés par les bons résultats qu'ils ont obtenu en DAP et leur robustesse au bruit. Pour la phase de classification, nous avons opté pour une architecture neuro-temporelle assez récente qui est l'architecture gamma très approprié pour la classification de phénomènes temporels.

2 Etude acoustique des consonnes emphatiques

Une analyse spectrale de la parole, distingue les voyelles des autres phonèmes par la présence d'énergie substantielle dans les régions de basses et moyennes fréquences, ce qui caractérise principalement les valeurs des trois premières fréquences formantiques appelées F1, F2 et F3.

Lors de la production d'une voyelle, le conduit vocal se déforme très peu, donc les fréquences des formants sont quasi-stables et spécifiques de la voyelle émise. Ce critère distinctif, basé donc sur la répartition de l'énergie qui est fortement liée aux formants, a permis l'introduction d'une classification acoustique des voyelles largement utilisée en DAP «les systèmes vocaliques » qui sont tous bâtis sur la double opposition grave/aigu, diffus/compact : le trait grave/aigu exprime la concentration de l'énergie dans les basses /u/ (respectivement moyennes et hautes /œ/ et /i/) fréquences et le trait diffus/compact exprime l'écartement important /i/ et /u/ (respectivement faible /œ/) de deux concentrations d'énergie (F1 et F2 séparés pour /i/ et /u/, rapprochés pour /œ/).

Il est à noter que l'étude des formants des voyelles nécessite non pas l'observation de fréquences précises mais de plages fréquentielles susceptibles de se recouvrir partiellement. Les deux premiers formants (F1, F2) des voyelles sont les plus intenses et, de ce faite, les plus importants pour leur vocalité, donc les plus informatifs. Cependant, dans la parole courante les fréquences formantiques varient étroitement en fonction de l'entourage phonétique de la voyelle que l'on cherche à produire.

Les consonnes emphatiques présentes dans la langue arabe possèdent les mêmes caractéristiques acoustiques que leurs variantes non emphatiques, seul leur contexte vocalique change, où les fréquences des formants (F1, F2) d'une voyelle qui suit une consonne emphatique dévient (figure 2) de leur position habituelle.

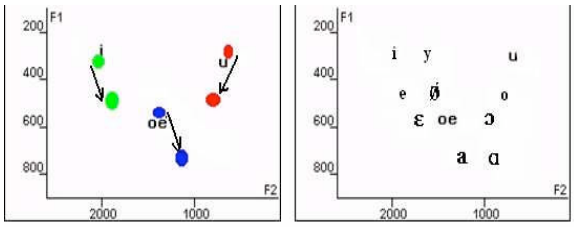


Fig. 2. Gauche: déviation des cibles formantiques (F1/F2) des voyelles qui suivent une emphatique. Droite : Cibles formantiques (F1/F2) de quelques voyelles présente dans la langue française.

En faisant l'analogie avec le système vocalique du français on constate que :

1. -Le /œ/ suivant une emphatique tend vers le /a/ (symbole API de «a») français
2. -Le /u/ suivant une emphatique tend vers le /o/ (symbole API de «o») français
3. -Le /i/ suivant une emphatique tend vers le /e/ (symbole API de «é») français

(API : Alphabet Phonétique International)

Donc afin de détecter une emphatique, nous avons conçu notre système de DAP afin qu'il distingue entre 6 voyelles : /œ/ («œ» en français), /a/ («a» en français), /u/ («ou» en français), /o/ («o» en français), /i/ («i» en français), /e/ («é» en français). Le système de DAP ne distinguera la consonne emphatique qu'après reconnaissance de la voyelle qui la suit [5], [6].

3 La parole, un phénomène temporel

La parole est un phénomène extrêmement variable dans le temps, la perception des phonèmes est difficile à établir voir impossible sur une courte durée (ex : un vecteur de 10 ms). Prendre une décision en négligeant cet aspect temporel de la parole peut entraîner beaucoup d'ambiguïtés. Une classification temporelle tire profit des informations sur le contexte (voisinage) afin de prendre une décision et peut donc résoudre une telle ambiguïté. Voici un exemple qui illustre une telle difficulté pour le cas des consonnes ص/س, ط/ت, ض/د. Vu leurs origines communes (alvéolaires), les caractéristiques acoustiques des phonèmes س, ص, ت, ط, د, ض présentent une structure de bruit similaire, seule leur mode d'élocution et le caractère emphatique de certains phonèmes changes. Pour ص/س le mode d'élocution est fricatif et apparaît dans le spectre comme un bruit assez long (environ 150 ms). La distinction entre les deux est basée sur le caractère emphatique de l'une par rapport à l'autre. Pour ط/ت le mode d'élocution est occlusif et apparaît dans le spectrogramme comme un silence suivi d'une explosion de bruit (Burst) qui a les mêmes caractéristiques acoustiques que ص/س mais la durée de ce bruit est plus petite (50 ms). Pour ض/د c'est le même mode d'élocution que ط/ت sauf que le Burst est précédé par un voisement dû à leur caractère sonore tout en ayant les mêmes caractéristiques acoustiques que le Burst de ط/ت.

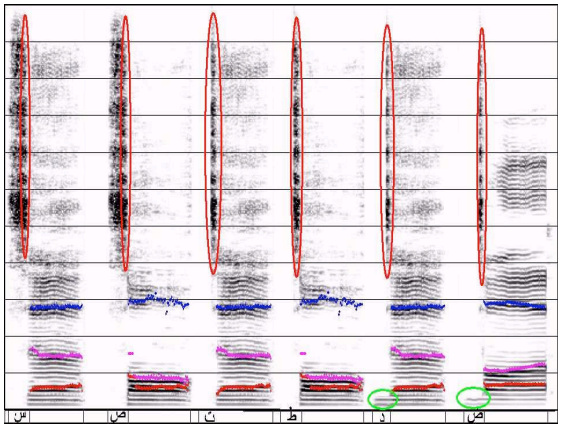


Fig. 3. Spectrogramme de ضَ / دَ /طَ / تَ /صَ / سَ, marqué, en haut un « Burst », en bas le voisement caractéristique des consonnes sonores.

4 Le codage MFCC

Le signal de parole est complexe et redondant et possède une grande variabilité. Pour qu'un système de DAP fonctionne efficacement, des informations caractéristiques et invariantes doivent être extraites du signal de parole. Cette procédure consiste à associer au signal de parole une série de vecteurs de paramètres généralement acoustiques. Le codage MFCC (Mel Frequency Cepstral Coefficients) est une extraction de caractéristiques du signal développée autour de la FFT (Fast Fourier Transform) et de la DCT (Discrete Cosine Transform), ceci sur une échelle fréquentielle non-linéaire (L'échelle Mel) [2], [3]. Ce codage paraît être le plus utilisé en DAP vus sa simplicité, sa robustesse au bruit et surtout aux très bons résultats en taux de reconnaissances qu'il a permis d'obtenir par rapport aux autres codages.

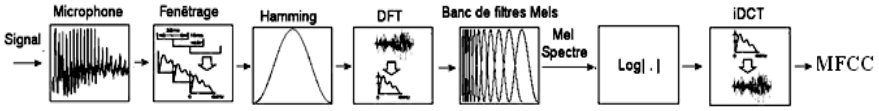


Fig. 4. Calcule des coefficients MFCC.

5 Les réseaux gamma

Le modèle gamma est une architecture assez récente bien qu'il puisse être très clairement apparenté à certains modèles comme le Time Delay Neural Network (TDNN). Le type d'architecture gamma permet de mettre en œuvre un réseau possédant deux types de mémoire du fait, d'une part, de l'organisation du réseau qui ressemble à un perceptron multicouche lorsqu'il est considéré à un niveau général, et, d'autre part, de l'organisation du neurone, qui ne respecte pas la définition habituelle d'un MLP.

Il est donc possible de considérer qu'il existe deux mémoires dont l'une est statique et située au niveau des connexions entre les neurones. Cette mémoire permet de conserver la connaissance définie lors de la phase d'apprentissage et ne variera pas en cours d'utilisation. Une deuxième mémoire, dynamique, est implantée dans le réseau au niveau des récurrences locales de certains neurones. Cette récurrence va permettre de conserver l'information pendant un certain temps, modélisant ainsi la dynamique de l'analyse du signal et, d'une certaine manière, l'état courant du système observé. Notre problème nous pousse à utiliser un système capable de modéliser à la fois des connaissances phonétiques instantanées, reposant sur les vecteurs d'entrée du réseau, et des connaissances phonétiques temporelles, reposant sur la durée antérieure de certains phénomènes. Nous avons donc décidé d'utiliser le mécanisme gamma comme base de notre architecture du fait de ses capacités et des connaissances mathématiques qui s'y rapportent. Le modèle gamma part d'une idée simple pour tenter de modéliser la décroissance exponentielle de l'activité. Un filtre est défini selon l'équation (1). Dans ce filtre, un seul paramètre, u, contrôle l'ensemble du

comportement et permet d'obtenir différents types de réponses de la part du neurone selon l'intervalle dans lequel se trouve ce paramètre.

$$y_{i,t} = u\, y_{i-1,t} + (1-u)\, y_{i,t-1} \quad \textbf{(1)}$$

Ce paramètre de feedback, tout comme les paramètres des « Autoregresive Networks », est limité sur un intervalle dans lequel la stabilité du filtre est démontrée. Les valeurs possibles du coefficient de régression sont, limitées à l'intervalle]0,2[. Au sein de cet intervalle, le filtre pourra avoir plusieurs comportements en fonction de la valeur de u. Ainsi, si u est égal à 1, la deuxième partie de l'équation disparaît et le filtre est alors équivalent à un délai simple. Les deux cas restants sont également très intéressants. Dans le cas où *u* est compris entre 0 et 1, bornes exclues, le filtre gamma se comporte comme un filtre passe-bas. Ce cas permet de simuler une mémoire à décroissance exponentielle et c'est celui qui nous intéressera. Seul l'intervalle]0,1] a, en effet, été étudié au cours de cette approche. Le cas restant, cas où *u* est compris dans l'intervalle]1,2[, permet au filtre de se comporter comme un filtre passe-haut.

Le filtre gamma, tel qu'il a été défini, n'est pas à proprement parler un neurone. Il s'agit d'un simple filtre dont l'utilisation première n'est pas pleinement de nature connexionniste. Il permet principalement de modéliser un système dynamique. Ce filtre peut être utilisé comme mécanisme de mémorisation au sein d'un réseau de type autorégressif, ce qui simplifierait l'équation et l'apprentissage dans ce dernier modèle. Mais le filtre gamma a tout d'abord été mis au point pour traiter et mémoriser le signal à un bas niveau. Il est donc destiné, en tout premier lieu, à améliorer les capacités du TDNN, ces réseaux étant très bien adaptés à la parole et largement répandus dans le domaine du DAP. Le mécanisme du filtre gamma permet d'effectuer des traitements qui sont impossibles à réaliser avec un TDNN sans une étape de prétraitement supplémentaire [1], [2].

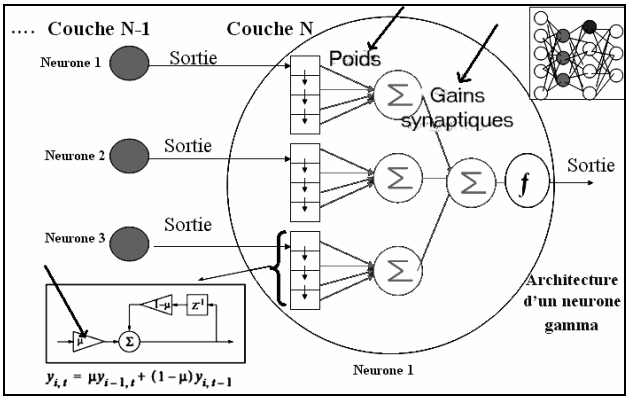


Fig. 5. Architecture d'un neurone gamma.

6 Notre base d'apprentissage

L'absence de base de données pour la langue arabe a constitué le principal problème rencontré dans notre travail, ainsi pour tester les performances de notre approche, nous avons construit notre petite base de données (une vingtaine de locuteurs ayant prononcé isolément en contexte consonantique emphatique ou non, les six voyelles établies précédemment œ, u, i, a, o, é) pour la langue arabe sous forme de fichiers « wav, 16bit, 22KHz, mono». Nous nous sommes heurtés à des problèmes qui sont principalement dus aux :

1. Défauts de prononciation de certains de nos locuteurs des phonèmes ذ / ض/ ظ
2. La mauvaise qualité de notre environnement et matériel d'acquisition
3. L'obligation d'étiqueter le corpus ce qui nécessite des connaissances en phonétique

7 L'étiquetage du corpus

La parole est un phénomène extrêmement variable. Le passage d'un phonème vers un autre n'est pas direct. Un phonème commence progressivement à perdre ses caractéristiques acoustiques à cause des mouvements articulatoires qui commencent à virer vers la position cible du phonème qui le suit. D'où l'existence de trois zones de transitions acoustiques en fonction de petites variations des paramètres articulatoires. Une première zone où le changement des paramètres acoustiques du premier phonème est moindre et presque stable. Une deuxième zone où le changement des paramètres acoustiques est brutal et où les caractéristiques des deux phonèmes se mélangent. Il est impossible d'affecter une telle zone à un des deux phonèmes ou de déterminer un début et une fin exacte aux deux phonèmes. Une troisième zone où les paramètres acoustiques du deuxième phonème se stabilisent [5].

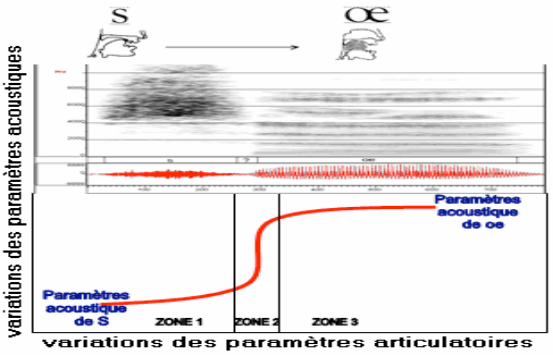


Fig. 6. Le changement des paramètres acoustiques de la transition phonétique /s/->/œ/ en fonction de variations des paramètres articulatoires.

Bien qu'il soit difficile de déterminer un début et une fin de phonème, les fricatifs et les voyelles montrent une certaine stabilité des caractéristiques acoustiques au centre du phonème, zone où les positions articulatoires cibles ont étaient atteintes. Nous exploiterons cette stabilité afin de positionner nos étiquettes pour les voyelles et les fricatifs (centre des phonèmes) car c'est là que la perception est maximale. Cette stabilité que l'on remarque chez les voyelles et les fricatives est due à la durée de ces phonèmes qui est assez importante, ainsi un locuteur peut rallonger cette durée à sa guise. Pour des plosives le mode d'élocution continu est impossible, leur production passe par des étapes très instables acoustiquement, mais essentielles à leur perception. Pour les plosives la perception ne se fait qu'à la fin de la zone instable (burst) et au début de la voyelle qui la suit. Cette position est idéale afin d'y déposer des étiquettes pour les plosives.

8 Le prétraitement

Après le chargement d'un fichier wav, le signal récupéré est fenêtré par une fenêtre glissante de 10ms de type rectangulaire qui avance avec un pas de 5ms, pour chaque fenêtre on calcule un vecteur de 16 coefficients MFCC. Si notre fenêtre appartient à un bloc étiqueté, on ajoute au vecteur acoustique une information sur l'identifiant du phonème étiqueté sinon on le marque comme appartenant à un bloc inconnu. On construit ainsi une base de données qu'on présentera au réseau gamma en faisant attention à l'ordre de présentation des vecteurs qui est très important (le temps).

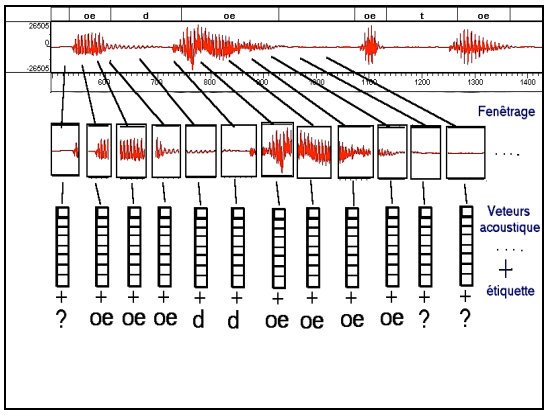


Fig. 7. L'extraction des vecteurs acoustiques à partir d'un signal avec étiquettes.

9 L'apprentissage

Pour l'apprentissage, on va faire agir le réseau gamma selon trois modes. Nous présenterons les vecteurs acoustiques à notre réseau selon leur ordre dans la liste. Un

vecteur acoustique non étiqueté mettra le réseau en mode écoute ce qui signifie que l'on va seulement propager le vecteur acoustique à travers le réseau sans récupérer la sortie ni corriger les poids. Ce bloc sera mémorisé par le réseau et est considéré porteur d'informations sur le phonème qui suit, mais pas assez pour le reconnaître. Le réseau restera dans cet état jusqu'à la rencontre d'un vecteur étiqueté. Si ce vecteur est destiné à l'apprentissage, le réseau se met en mode apprentissage, propage le vecteur acoustique à travers le réseau, calcule l'erreur de sortie et ajuste ses paramètres afin de minimiser l'erreur de sortie. Enfin, si le vecteur est destiné au test, le réseau se met en mode test. Il propage le vecteur dans le réseau, récupère la sortie, la compare à l'étiquette afin de déterminer le taux de reconnaissance.

10 La reconnaissance

Après avoir chargé un fichier « wav », la reconnaissance d'un phonème passe par le calcule des vecteurs acoustiques et leurs propagations dans le réseau gamma. On ignore les vecteurs à énergie faible (exemple < 20dB), ceux dont le pic de sortie est faible et ceux dont la durée est très courte (exemple < 50 ms) qui sont supposés bruit (parasites). La prise de décision se fait à chaque entrée, donc on aura autant de sorties que d'entrées. Une sortie possible pour / سُ صُ / est : *"_ _ _ _ sssssssuuuuuuu_ _ _ sssssssooooooo_ _ _ _"*.

On représente les segments ignorés par des /_ / qui seront remplacés par un espace, après regroupement on obtient la transcription / *su* so /. L'apparition de parasites due au bruit n'est pas à négliger, on pourra envisager une étape de lissage précédant l'étape de regroupement.

11 Application

Afin d'évaluer les performances de notre approche, nous avons effectué nos tests sur des enregistrements au format « wav, 22KHz, 16bits, mono ». Puisque l'apprentissage du réseau gamma est de type supervisé, les échantillons d'apprentissages et de tests doivent être étiquetés manuellement. Notre base d'apprentissage comporte une vingtaine d'échantillons pour chaque variante de consonne emphatique ou non. Le réseau gamma que nous avons utilisé, comporte 16 neurones en entrées qui correspondent à la dimension des vecteurs MFCC utilisés, 16 neurones en couche cachée fixés empiriquement, et 12 neurones en sorties qui correspondent au nombre de classes considérées pour 6 consonnes ayant une variante emphatique plus les 6 voyelles considérées précédemment. La taille de la mémoire des neurones du réseau gamma a été fixée empiriquement à 5. L'apprentissage à été effectué sur un « Pentium4, 2.8GHz, 256 Mo de RAM ». Le réseau gamma permet une reconnaissance en un temps record qui ne dépasse pas la seconde. La figure 8 montre un exemple de résultats obtenus après reconnaissance par notre réseau gamma.

On constate que le réseau gamma arrive très bien à distinguer la voyelle « œ » de la voyelle « a » qui sont généralement confondues dans la langue arabe. Cette distinction nous permet d'affirmer le caractère emphatique ou non d'une consonne. Ainsi dans

cet exemple un « s » suivi de « œ » est reconnu comme une réalisation non emphatique «سا ». Même chose pour le « s » suivi de « a » qui est reconnu comme une réalisation emphatique «صا ».

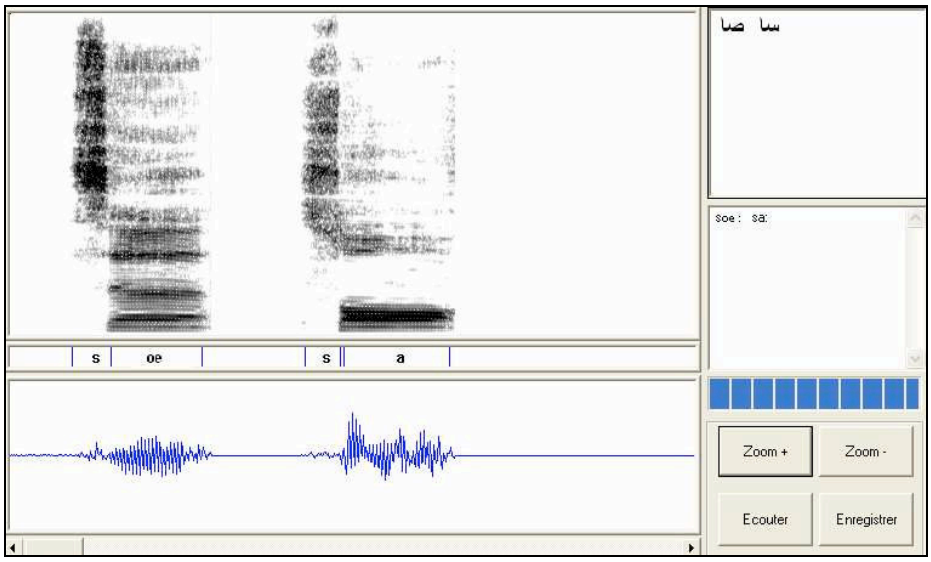


Fig. 8. Exemple de distinction entre emphatique et non emphatique par notre système de décodage acoustique phonétique.

12 Conclusion

Dans ce travail nous avons présenté un système de décodage acoustique phonétique de la langue arabe, capable de distinguer des consonnes emphatiques de leurs variantes non emphatiques. Notre approche repose sur la classification par un réseau gamma de vecteurs MFCC, extrais d'un petit corpus que nous avons soigneusement étiqueté. La construction de ce corpus représente la principale difficulté dans ce travail et surtout son étiquetage qui demande beaucoup de connaissances en phonétique, et des heures de travail.

L'aspect emphatique ne constitue pas le seul problème dans un système de reconnaissance de la parole pour la langue arabe. D'autres problèmes existent, comme la détection de la gémination connue sous le nom de « chedda » qui est primordiale, car son apparition peut changer complètement la signification d'un mot de la même façon qu'une consonne emphatique.

Néanmoins, toute étude sur la langue arabe ne restera que sur le plan théorique, tant qu'il n'y aura pas de réelle base de données pour l'arabe standard et dialecte, faites par des experts phonéticiens. Cela ne réussirait qu'à donner plus de crédibilité aux travaux concernant la langue arabe et permettra de comparer les différents résultats obtenus.

References

1. Steve Lawrence, Ah Chung Tsoi, Andrew D. Back: The Gamma MLP for Speech Phoneme recognition. Advances in Neural Information System, vol.8 (1996) 785-791
2. Laurent Buniet : Traitement automatique de la parole en milieu bruité, étude de modèles connexionnistes statiques et dynamiques. Thèse présentée et soutenue publiquement le 10 février 1997 pour l'obtention du Doctorat de l'Université Henri Poincaré – Nancy 1 spécialité informatique, France
3. Nguyen Quoc Cuong : Reconnaissance de la parole en langue Vietnamienne. Thèse présentée et soutenue publiquement le 19 juin 2002 pour l'obtention du grade de Docteur de l'INPG de l'Institut National Polytechnique de Grenoble, France
4. Bruno GAS : Méthodes neuronales pour l'extraction de caractéristiques non linéaires et discriminantes, application aux signaux de parole. Habilitation à Diriger des Recherches présentée le 23 novembre 2005 à l'Université PARIS VI, France
5. Kenneth N. Stevens: On the quantal nature of speech. Journal of Phonetics vol.17 (1989) 3-45
6. Neagu Adrian : Représentations phonétiques et identification des syllabes occlusive voyelle en français. Thèse présentée et soutenue le 30 avril 1998 pour l'obtention de grade de Docteur de l'Institut National Polytechnique de Grenoble.
7. Tahar Saidane, Mounir Zrigui, Mohamed Ben Ahmed : La transcription orthographique phonétique de la langue arabe. Récital, Fès, 19-22 avril 2004.

Application des algorithmes génétiques multi objectif Au problème d'exploration Des espaces de configurations des systèmes embarqués

*A.Bengueddach, **S.Niar, *B.Beldjilali

*Département informatique, Faculté des sciences
Université d'Oran Es-Senia, B.P 1524 Oran El Mnouer

** Université de Valenciennes et du Hainaut-Cambrésis - ISTV2- Le Mont Houy
59313 VALENCIENNES Cedex 9
bengueddachasmaa@yahoo.fr, smail.niar@univ-valenciennes.fr, bouzianebeldjilali@yahoo.fr

Résumé. Les systèmes complets intégrés sur une seule puce de silicium, ou System On Chip (SOC) peuvent contenir différents composants hétérogènes. Cette évolution fait que la conception de ces derniers devient très complexe. A cela s'ajoute un besoin sans cesse grandissant de systèmes flexibles, à faible consommation énergétique, performants, respectant le temps de mise sur le marché et à coût minimal. Face à cette problématique, il est nécessaire de connaître toutes les optimisations possibles à tous les niveaux de conception. On s'intéresse dans ce papier à l'optimisation au niveau architecture et aux choix des paramètres optimaux pour le cas de processeur super Scalaire. Cette technique tient compte de deux critères : performance et consommation d'énergie. Le problème devient alors une optimisation multi objectifs basée sur des algorithmes génétiques multi objectifs NSGAII. Les résultats sont obtenus par simulation en utilisant des benchmarks de taille significative représentant les applications cibles. L'ensemble optimal au sens de Pareto définira l'ensemble des solutions à notre problème.

Mots clés : SOC, Algorithme génétique, optimisation multiobjectif, NSGA-II, exploration de l'espace de conception, estimation des performances.

1 Contexte : Conception des systèmes embarqués

1.1 Définitions

Le terme « système embarqué » désigne tout sous-système spécifique à un domaine d'application et devant être intégré dans un système plus grand tel qu'un produit, un instrument ou un autre système[1].

Un système embarqué est un système mixte souvent autonome, contenant du matériel et du logiciel et qui est complètement intégré dans l'environnement qu'il contrôle [2].

L'interaction avec l'environnement se fait grâce aux capteurs analogiques couplés à des convertisseurs A/N qui permettent d'avoir des informations en entrée. En sortie des actionneurs généralement analogiques couplés à des convertisseurs N/A, communiquent des informations traitées par le calculateur mettant en œuvre un processeur embarqué et ses périphériques coprocesseurs à l'environnement extérieur [3].

1.2 Importance du marché de l'embarqué

Les systèmes embarqués sont présent dans de nombreuses applications et différents domaines, par exemple [4] dans les systèmes orientés contrôles ou systèmes temps réel dur (système de guidage embarqué pour les automobiles [5], développement des systèmes à hyperfréquence de guerre électronique embarqués sur avion de chasse : brouilleurs, détecteurs anti-missiles, systèmes de contre mesure). D'autres secteurs importants tels que les systèmes orientés calcul et traitement du signal.

La conception de l'électronique embarquée (logiciel et matériel) représente 20% du coût de conception des automobiles et des avions. Ce chiffre atteint 50% pour certains appareils ménagés et il est prévu que ce coût dépasse les 80% dans les cas des téléviseurs et autres appareils électroniques à large public [6].

Le domaine de l'automobile comprend de très riches fonctionnalités tant complexes que diversifiées (Tableau1). C'est là où on a les systèmes embarqués les plus nombreux.
La conception de tels véhicules fait intervenir de nombreuses technologies :

- La technologie « by-wire » : les systèmes mécaniques de transmission sont remplacés par des bus électroniques contenant des unités de calcul, capteurs et actionneurs.
- D'autres évolutions technologiques comme l'introduction du logiciel (téléchargement du logiciel pour réparation ou évolution).
- Différents réseaux sont utilisés sur le véhicule, par exemple un réseau CAN faible débit pour connecter les calculateurs de gestion des portes, l'éclairage et la visibilité ; un autre réseau haut débit MOST connectant les applications de communications, multimédia et Internet.

Catégorie	**Nouveau service**
Sécurité	Assistance au freinage (ABS)
Moteur	Injection électronique
Confort	CD Radio
Navigation	Suivi de parcours par GPS
Navigation	Repérage et guidage sur voie
Navigation	Contrôle de la vitesse
Productivité	Gestion de flotte : repérage, aide à la navigation, informations professionnelles
Sécurité	Airbags intelligents
Information	Connexion Internet (UMTS)
Sécurité	Assistance anti-collision
Confort-Sécurité	Commandes vocales
Confort	Direction tout électronique (steer-by-wire)
Sécurité	Freinage tout électronique (brake-by-wire)
Confort	Assistance au stationnement
Confort Sécurité	Ecran de projection

Tableau1 : Nouveaux services dans l'automobile [7]

1.3 Intégration des systèmes embarqués sur une seule puce

Dans un rapport publié par l'ITRS [8] annonce que 70% des ASIC (Application Specific Integrated Circuit) en 2005 comportaient au moins un CPU embarqué. Ainsi, la technologie de fabrication des circuits intégrés a permis de passer des composants spécifiques ASIC aux systèmes embarqués sur une seule puce SOC pour « System On Chip ».
De plus, d'autres technologies très sophistiquées non digitales (analogique ou RF) sont intégrées sur ces puces.

Le tableau 2 illustre quelques exemples de produits mono puce. Des composants spécifiques (DSP, MCU) sont utilisés pour implémenter un contrôle et/ou un traitement spécifique. Différents types de réseaux sont employés (bus hiérarchiques, réseaux de commutateurs, etc.). De plus mémoires et processeurs sont inclus dans cette architecture.

Il existe de nombreuses raisons pour lesquelles l'intégration sur une seule puce représente une solution architecturale pour implémenter un système [10] :

- Des processus de fabrication de plus en plus fins permettant la combinaison sur une seule puce de la logique et la mémoire, ce qui permet d'optimiser le temps global d'accès mémoire.
- Comparée à un système sur carte, la durée de conception est courte permettant ainsi une mise sur le marché rapide.

Domaine D'application	**Traitement des données**	**Réseau de comm.**	**Exemple typique**
Terminaux télécoms	1 DSP	Bus hiérarchiques	VDL (ST)
Multimedia	Quelques DSP	Réseau de communication	TRIMEDIA (Philips)
Processeurs de réseau	Beaucoup de DSP	Réseau	IXP2800 (Intel)
Processeurs de jeux	Quelques DSP	Réseau hiérarchique	PlayStation 2 (Sony)

- Une faible consommation permet d'avoir des produits plus économiques et plus performants.

Tableau 2 : Etat de l'art sur les produits multiprocesseurs mono puce [9]

1.4 Etat de l'art sur les travaux de conception

Ce travail s'inscrit dans le contexte de la conception des systèmes embarqués mono puces. Aujourd'hui, le choix d'une architecture n'est pas trivial et il faut trouver un compromis entre flexibilité, consommation, performance, coût et la rapidité de conception en terme de TTM (Time To Market). Les recherches dans la conception architecturales des SoC ont pris deux axes différents.

1.4.1 Architecture monoprocesseurs mono puce

Le but de cette première tendance est d'améliorer considérablement les performances du CPU par l'usage de co-processeurs. Un tel CPU se caractérise par une fréquence très élevée, des structures matérielles spécialisées, un ensemble d'instructions sophistiquées, plusieurs niveaux de hiérarchie mémoire (plusieurs niveaux de caches), et des techniques spécialisées d'optimisation logicielle (nombre d'accès mémoire, taille, etc). On peut citer comme exemple : Power PC, Intel Pentium 4, ST 100, etc.

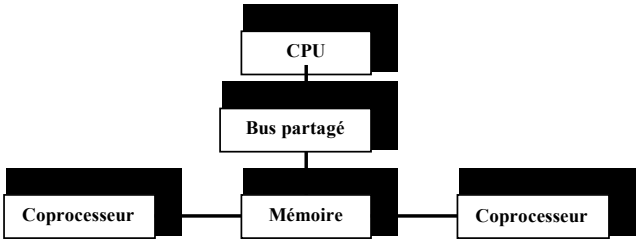


Figure 1 : Un exemple d'architecture monoprocesseur mono puce [09].

Dans ce type d'architecture (figure 1), la communication est basée sur le principe maître/esclaves ; le CPU est le maître et les périphériques sont les esclaves. Les communications (par interruptions) se font généralement via un bus partagé. Le processeur est donc responsable du calcul aussi bien que la communication qui engendre souvent une dégradation des performances.

1.4.2 Architecture multiprocesseur mono puce (AMM)

La deuxième tendance s'intéresse aux multiprocesseurs (multi maîtres). Les principales raisons sont [9] :

- Le parallélisme : Afin d'effectuer plus de travail par cycle d'horloge, un AMM évite la limitation des processeurs super scalaires (cas des instructions interdépendantes) en employant le parallélisme au niveau tâche. Ce type de parallélisme est obtenu en exécutant simultanément des séquences d'instructions complètement séparées sur chacun des processeurs.
- Les retards causés par les interconnexion : Comme le portes COMOS deviennent plus rapides et les puces deviennent physiquement plus grandes, le retard accumulé par les interconnexions entre les portes devient significatif. Cependant, pour remédier à ça les processeurs occupent un espace réduit sur la puce, réduisant ainsi au minimum la longueur de ses fils.
- Le temps de conception : Un AMM permet de réduire le temps de conception car il permet de réutiliser des composants pré validés dans plusieurs applications de différentes tailles.
- Cette architecture est structurée en couches (figure2). La partie matérielle est composée de deux couches [08] :
- La couche basse contient des composants standards utilisés par le système tels que les processeurs DSP, MCU, IP et mémoire.
- La couche de communication matérielle embarquée sur la puce peut contenir un réseau de communication allant du simple pont jusqu'au complexe (communication de paquets).
- Le logiciel embarqué est aussi découpé en couches :
- La couche basse contient les pilotes d'entrée/sortie et d'autres contrôleurs de bas niveau permettant de contrôler le matériel.
- Une couche de gestion de ressources permet d'adapter l'application à l'architecture. Elle fournit les fonctions nécessaires à l'application. Cette tâche peut être fournit par la plupart des systèmes d'exploitation (OS) mais pour des raisons de taille et/ou de performance, il est préférable de réaliser une couche spécifique pour l'application.
- Le code de l'application.

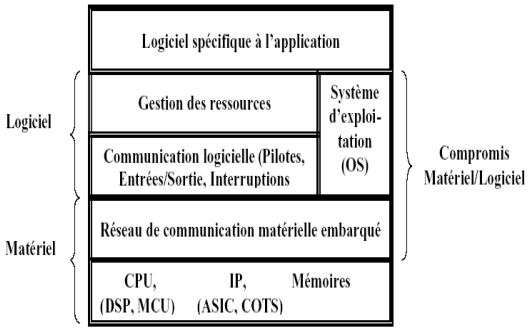


Figure 2 : Représentation d'un système embarqué sous forme de couches

1.5 Processus de conception de systèmes sur puce

La majorité des méthodes de conception actuelles copie le flot de conception des cartes. Le processus de conception est décomposé en quatre métiers, l'architecture système, la conception des composants matériels, la conception des composants logiciels et l'intégration ou la conception système (figure 3.a)

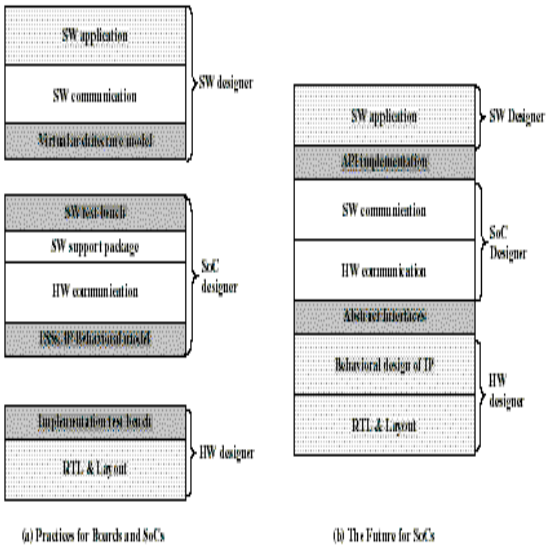


Figure 3: Les métiers de la conception système [8]

Tout le matériel est taillé sur mesure : les composants et le réseau de communication. Le logiciel est réalisé à part par une autre équipe. La vérification des différentes parties est aussi réalisée de manière séparée et chaque équipe utilise des programmes de tests spécifiques. Le résultat de cette séparation (logicielle, matérielle) peut entraîner des surcoûts.

Cependant ce schéma n'est plus applicable à partir d'un certain degré de complexité.

La conception d'architecture consistera surtout à assembler des composants existants en vue de respecter des contraintes de performances et de coûts au niveau du système global et non au niveau du composant. Ainsi, il convient de combiner la conception des couches de communication logicielles et matérielles (figure 3.b).

Concernant cette approche de conception, il faut définir un flot partant de la spécification système à la réalisation au niveau RTL. Un flot de conception complet est constitué de deux étapes :

L'exploration d'architecture exploite/utilise la description de l'application au niveau système, et explore par l'assistance d'un outil d'estimation des performances, l'espace de solution architecturale pour trouver l'architecture système optimale pour l'application spécifique [9]. Les contraintes de temps d'exécution, de surface, de consommation, et de coût doivent être prises en compte durant cette étape. A la fin de la phase exploration d'architecture les paramètres architecturaux (optimaux) dédiés à l'application sont fixés. Ces paramètres sont utilisés dans la seconde étape.

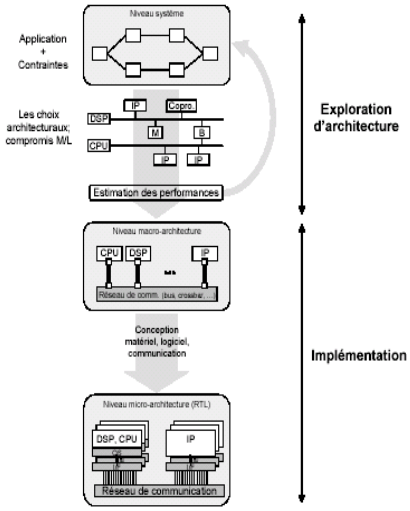


Figure 4: Flot du système au RTL

L'étape d'implémentation pour produire l'architecture RTL comporte : la conception des modules matériels et la conception du réseau de communication permettant d'intégrer les composants de base. Cette étape est réalisée de façon automatique basée sur l'instanciation et la configuration des composants dans une bibliothèque.

Ce flot considère trois niveaux d'abstraction (figure4): le niveau système, le niveau macro architecture et le niveau micro architecture.

Au niveau système le circuit est spécifié, les différents modules sont décrits en utilisant des langages. Ces derniers peuvent communiquer avec des protocoles de haut niveau. La co-simulation est utilisée pour valider la fonctionnalité du modèle.

Le second niveau représente une architecture abstraite qu'on appel macro architecture, ce modèle est composé des module représentant des processeurs liés par des fils logiques. Le processeur peut être un DSP ou un microcontrôleur exécutant le logiciel, un processeur matériel ou un IP existant (mémoire globale, périphériques, contrôleur de bus, etc). Les fils logiques sont des canaux abstraits qui transfèrent des données fixes (entier, flottant). Les différents modules peuvent être décrits en utilisant un ou plusieurs langages. La co-simulation peut être utilisée pour valider le découpage architectural.

Le dernier niveau est le niveau RTL (ou microarchitecture), tous les détails de la communication entre les composants sont réalisés. Les couches logicielles peuvent comporter un système d'exploitation spécifique (OS). Les couches matérielles comportent les bus et autres dispositifs physiques permettant d'acheminer les informations entre les composants.

1.6 Problèmes liés à la conception des systèmes embarqués

La conception de ces systèmes rend difficile la mise en œuvre des méthodes de conception existantes. Les principaux défis sont [8], [10] :

- La conception des systèmes multiprocesseurs mono puce implique la réutilisation de composants existants. Il sera donc nécessaire d'utiliser des méthodes d'assemblage de composants hétérogènes.
- Le concepteur d'un système contenant des processeurs programmables doit obligatoirement fournir en plus du matériel, une couche logicielle permettant la programmation de haut niveau du circuit.
- L'assemblage de composants hétérogènes sur une même puce peut nécessiter des schémas de communication très sophistiqués mettant en œuvre des parties matérielles, des parties logicielles et des parties non numériques (interfaces optiques ou micromécaniques).

- la complexité des systèmes au niveau RTL décrit en détail (niveau cycles horloge), rend difficile la validation manuelle. Des méthodes de conception /validation de haut niveau sont nécessaire.
- Un problème majeur dans la conception de ces systèmes est la maîtrise de la consommation, du point de vue de la puissance dissipée qui génère des problèmes thermiques et la nécessité d'utiliser des dispositifs de refroidissement coûteux, que du point de vue de l'énergie due à des besoins croissants en puissance de calcul et une évolution limitée des possibilités des batteries.
- De plus, la flexibilité croissante demandée par les applications augmente l'importance des méthodes dynamiques de gestion logicielle, notamment par la modification de paramètres à différents niveaux : mode de fonctionnement, tension d'alimentation variable, architecture reconfigurable, mises en veille, etc.
- Dans un rapport de l'ITRS, le message énoncé est : « Le coût de la conception constitue la plus grande menace à l'avenir de l'industrie des semi-conducteurs.
- La maîtrise de la conception des systèmes mono puce complexes devient un facteur crucial et prioritaire tout en respectant les contraintes de mise sur le marché et les objectifs de qualité.

1.7 Optimisation des ressources pour la conception des SoC

Face aux problèmes posés dans le paragraphe précédent, et pour concevoir efficacement le système, il faut connaître à tous les niveaux de conception, les possibilités réelles d'optimisation des différents paramètres en terme de puissance, d'énergie, de performances et de flexibilité [11] :

- **Application :** Description des contraintes de consommation selon les différents types d'application, définition et utilisation des IP, méthodes et solutions proposées dans l'industrie, etc.
- **Système :**Techniques d'ordonnancement, de partitionnement, OS, asynchrone, etc.
- **Algorithme :** Optimisation de code, estimation de logiciel embarqué, etc.
- **Architecture :** Architecture parallèle, pipeline, reconfigurable, analogique, etc.
- **Logique :** Optimisation des opérateurs, asynchrone, etc.
- **Technologique :** Possibilités actuelles de variation de la tension d'alimentation et de la tension de seuil, impact de ces variations sur les niveaux supérieurs, effet des différentes technologies sur le flot de conception classique, etc.

2 Algorithmes évolutionnaires mono et multi objectifs

Les concepteurs sont confrontés de plus en plus à des problèmes de complexité croissante [2]. De nombreux secteurs tels que la recherche opérationnelle, la conception de systèmes sur puce, etc. tentent de maîtriser cette complexité. Pour résoudre le problème, on le considère souvent comme un problème d'optimisation complété par de contraintes.
Une fonction d'objectif (ou plusieurs) est définie et on cherche à optimiser par rapport à tous les paramètres concernés.

2.1 Optimisation mono objectif

Dans le cas d'un objectif unique, la définition de la fonction d'adaptation ne pose généralement pas de problème. Par exemple, si l'on se fixe l'objectif de trouver une configuration du processeur dont la durée d'exécution est minimale, cette fonction sera la durée d'exécution de l'application sur le processeur [2].

2.2 Optimisation multi objectif

Certains problèmes d'optimisation doivent viser des objectifs multiples, souvent contradictoires, ce qui nécessite un compromis. La méthode la plus simple consiste à définir plusieurs fonctions objectifs f_i, chacune correspond à un objectif à atteindre, et à les combiner par une sommation pondérée [10]:

$$f = \sum_i \alpha_i f_i \qquad (1)$$

Où les α_i sont des coefficients poids positifs dont la somme vaut 1.
Au lieu de la somme, on peut utiliser un produit :

$$F = \prod \alpha_i f_i \qquad (2)$$

D'autres méthodes « non agrégatives » peuvent être utilisés, les plus connus :

- La méthode Vector Evaluate Genetic Algorithm « VEGA ».
- La méthode Multiple Objective Genetic Algorithm « MOGA ».
- Pareto Archived Eolution Stratey « PAES ».
- Strength Pareto Evolutionary Algorithm « SPEA ».
- La méthode Niched Pareto Genetic Algorithm « NPGA».
- La méthode Non dominated Sorting Genetic Algorithm « NSGA »

Dans le cadre de ce travail, nous avons étudié en détail la version NSGA-II de la méthode NSGA.

2.3 L'algorithme NSGA-II

La base de la l'algorithme NSGA-II est la classification des individus. Tous les individus non dominés, de même rang sont classés dans une catégorie, à laquelle on affecte une efficacité qui est inversement proportionnelle au rang de Pareto de la catégorie considérée [10].

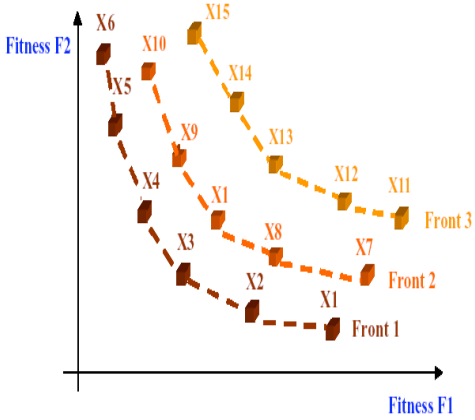


Figure5 : Classification des individus en plusieurs catégories suivant le rang de Pareto

L'algorithme NSGA-II utilise la technique crowding distance pour préserver la diversité des solutions sur le front de Pareto. Cette procédure s'applique sur le dernier front de Pareto pour compléter la taille de la population parents pour la génération suivante.

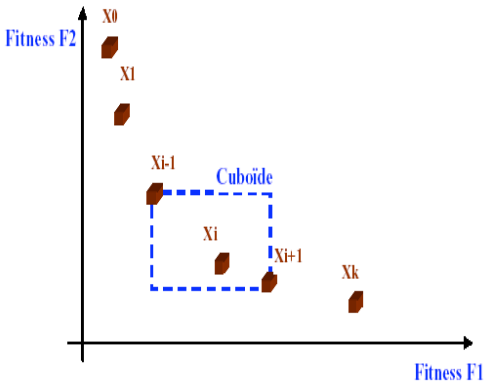


Figure 6 : Diversité dans la représentation des individus

La boucle principale de l'algorithme crée aléatoirement la population initiale P_0. Chaque individu de cette population est affecté à un front de Pareto adéquat. On applique les opérateurs génétiques (sélection, croisement et mutation) pour générer la population enfant Q_0 de taille N à partir de la population P_0.

Début
Rt <- Pt U Qt /*Combinaison des pop. parent et enfant |Rt| = 2N*/
F <- fast-nondominated-sort(Rt) /*F = {F1, F2,…} ensemble des front de Pareto tel que $\sum |F_i| = 2N$ */
Pt+1 <- ϕ /*initialisation*/
i <- 1
Tant que |Pt+1| + |Fi| ≤ N /*jusqu'à la taille max de la pop. N*/
Crowding-distance-assignment(Fi) /*calcul de la distance «crowded» dans Front Fi*/
Pt+1 <- Pt+1 U Fi /*Reconstruction de la population parent en ajoutant ième front de Pareto*/
i <- i + 1 /*incrémentation vers le front suivant*/
Fin tant que
Sort(Fi, <n) /*Triage des solution Fi dans l'ordre décroissant suivant <n */
Pt+1 <- Pt+1 U Fi[1:(N - |Pt+1|)] /*Choisir juste les premiers éléments Fi pour compléter la population
parent Pt+1 jusqu'à N individus*/
Qt+1 <- make-new-pop(Pt+1) /*Application des opérateurs génétique (sélection, croisement, mutation) pour
la génération d'une nouvelle pop. Qt+1*/
t <- t + 1 /*génération suivante*/
Fin

Algorithme 1 : Boucle principale NSGA-II

On distingue quatre étapes :

- Génération d'une population enfants Qt à partir de la population parents Pt par l'application des opérateurs génétiques (croisement et mutation). Cette manipulation est suivie par la fusion de deux populations enfants et parents dans une seule population globale Rt.
- Classement des individus de la population globale Rt en plusieurs front de Pareto dans l'ordre croissant (F1, F2, F3, etc.).
- Reconstruction de la population parent pour la génération suivante (Pt+1) par la sélection d'individus appartenant aux premiers fronts de Pareto (F1, F2 et F3) tels que :

$$|P_{t+1}| = \sum_{k=1}^{j} |F_k| \leq N \qquad (3)$$

N : La taille de la population, *j* : Nombre de fronts sélectionnés et
|Fk| : Nombre d'individus dans le front de Pareto d'ordre k

- Application de la technique de *crowding distance* sur le front de Pareto Fj+1 (c'est F4 dans le cas de la figure 3.13) pour compléter la population Pt+1 à une taille de N. Pt et Qt, sont respectivement la population parent et enfant de taille N. La nouvelle population Pt+1 de taille N est utilisée avec l'application des opérateurs génétiques pour la génération d'une nouvelle population Qt+1 de taille N.

2.4 Implémentation de NSGA-II

Les algorithmes génétiques (AGs) et le recuit simulé (RS) sont deux techniques d'optimisations stochastique travaillant sur le même type de problèmes. Il est logique de chercher à les associer afin de tirer parti de leurs avantages respectifs. Plusieurs travaux ont montré que le RS converge généralement plus vite vers la solution optimale lorsque le problème est de taille raisonnable (à l'inverse pour les AG) alors associer les deux techniques améliore davantage la convergence des AGs [12].

Il existe différentes manière pour faire la sélection des individus, les plus connus sont : par roulette ou encore la sélection par reste stochastique sans remplacement (et bien d'autres).

Le problème majeur de ces sélections classiques est qu'on peut facilement tomber dans le cas suivant : On voit (figure7) qu'il y a un élément proche de l'optimum local M2 et dont la fonction d'adaptation est bien plus élevée que celle de tous les autres éléments. Cet élément risque d'être le seul représentant pour la génération suivante, alors que d'autres éléments plus proches de l'optimum global M1 disparaîtront. Cet optimum ne pourra être ensuite atteint que très difficilement au prix de nombreux essais successifs.

Les solutions à ce problème : Scaling, sharing, permettent de conserver à la population (pop) une diversité pour l'empêcher de se concentrer trop vite autour des meilleurs éléments, proches le plus souvent d'un optimum local.

Une telle concentration prématurée de la population permet très difficilement d'atteindre ensuite un meilleur optimum local (sachant que le but est d'atteindre ou du moins s'approcher de l'optimum global). Une autre solution pour rendre la sélection moins radicale au début du déroulement de l'algorithme (on parle de pression sélective) et de permettre à la pop de se diversifier (faire un croisement), et là on pourra introduire des techniques issues du recuit simulé.

Le principe de croisement par le Recuit Simulé : on commence par sélectionner 2 parents P1 et P2 dans la pop.

On applique ensuite l'opérateur de croisement classique pour obtenir C1 et C2.Un tournoi est alors effectué entre le parents et les enfants dont les 2 vainqueurs sont sélectionnés par le schéma de recuit comme suit :

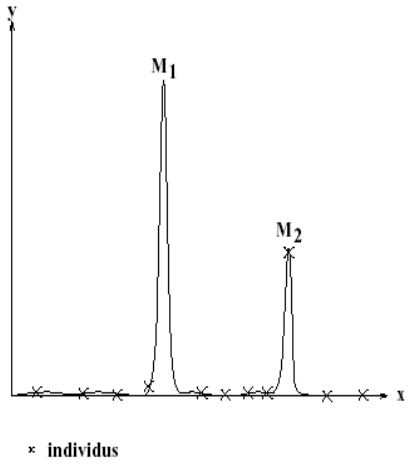


Figure 7 : Exemple où les sélections classiques risquent de ne reproduire qu'un individu

- On considère l'individu C1.
- On tire ensuite aléatoirement un des 2 parents : soit Pi ce parent si C1 est meilleur que Pi alors C1 est sélectionné sinon C1 est sélectionné avec la proba :

$$P = e^{-\left(\frac{|f_{C_1} - f_{P_i}|}{c[n]}\right)} \quad (4)$$

Où : c(n)est un coefficient décroissant de la génération courante n.

- On fait la même chose pour C2 avec le parent restant.
- On obtient C'1et C'2.

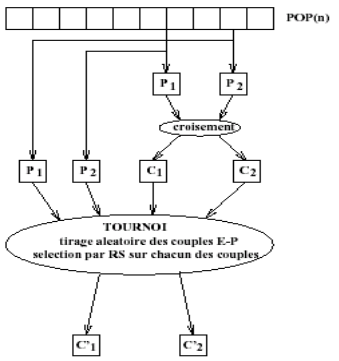


Figure 8 : Principe du croisement avec recuit simulé

3 Exploration de l'espace architectural à base de simulation

L'exploration architecturale est une étape du flot de conception des systèmes embarqués complexes. Elle consiste essentiellement à définir : l'architecture, le nombre et le type de ressources ou de composants (d'exécution, de communication et de mémorisation).

Pour explorer l'espace des solutions, il est nécessaire d'exécuter l'ensemble des fonctionnalités du système tout en respectant les contraintes de performances, taille mémoire ainsi que la date d'arrivée sur le marché.

Il s'agit d'implémenter le système (l'application SoC) à concevoir sur un ou plusieurs simulateurs [10]. Ceci permet au concepteur de vérifier le bon fonctionnement de son application à savoir l'estimation des performances sans passer par une réalisation réelle du système à étudier. On s'intéresse dans notre travail aux simulateurs SimpleScalar et Sim-cache pour configurer le cache.

3.1 Evaluation des performances

3.1.1 Motivations et objectifs

L'exploration de l'espace des solutions architecturales est une phase importante dans le procédé de conception des systèmes complexes. Elle consiste à trouver la meilleure architecture du système, la détermination des protocoles de communication, la topologie du réseau de processeurs et le placement/ordonnancement. Pour un système de n processus, le nombre de partitions différentes que l'on peut construire par l'équation [9]:

$$U_n = \sum_{q=1}^{n} \sum_{i=0}^{q} \frac{(-1)^{q-1}(i)^n}{(q-i)!(i)!} \qquad (5)$$

Soit p processeurs différents (matériels/logiciels), le nombre d'architectures possibles est obtenu par l'équation suivante :

$$V_n^p = \sum_{q=1}^{n} p^q \sum_{i=0}^{q} \frac{(-1)^{q-1}(i)^n}{(q-i)!(i)!} = \sum_{i=1}^{n} i^n \frac{p^i}{i!} \sum_{f=0}^{n-i} \frac{(-p)^f}{f!} \qquad (6)$$

Le nombre d'architectures possibles varie en fonction du nombre de processus et le nombre de types de processeurs. Par exemple, pour un modèle composé de 10 processus et une technologie comportant trois types de processeurs (deux types de microprocesseurs pour le logiciel et un pour le matériel) l'espace de solutions contient $4{,}872.10^7$ éléments. Le nombre accroît exponentiellement avec n et p.

Le besoin d'un outil capable d'estimer les performances rapidement, pour effectuer cette tâche d'exploration d'architectures est fondamental.

3.1.2 L'état de l'art concernant l'estimation de performance

L'évaluation des performances est devenu un enjeu économique pour la conception d'un système sur puce. En effet, construire un système sans l'analyse de ces performances mènera soit à un système sous dimensionné inutilisable, soit à un système surdimensionné qui entraînera un gaspillage financier [10].

On se base sur le nombre de cycles d'horloge du processeur nécessaire à l'exécution d'une application pour l'estimation des performances. Le nombre de cycles est déterminé par l'utilisation d'un simulateur (SimpleScalar), car les processeurs actuels utilisent des techniques très évoluées. Cette complexité fait qu'il est difficile de les tester et de comprendre leur fonctionnement [12]. En plus, il est quasiment impossible de modifier ou tester de nouveaux concepts directement sur ces derniers.

Les méthodes d'estimation sont classées dans trois catégories : statiques, dynamiques et mixtes [12]:

- Dynamique : les mesures de la performance d'une architecture sont le résultat de l'exécution d'un modèle (par simulation).
- Statique : l'estimation de la performance d'une architecture est le résultat d'une analyse statique d'une spécification (analyse de chemins dans une spécification de flot de contrôle).
- Mixtes dynamique/statique: se base sur l'utilisation des deux approches précédentes pour l'analyse de performance d'une architecture.

Les approches dynamiques sont très précises mais leur principal inconvénient est le temps nécessaire pour l'obtention du modèle à simuler et le temps de la simulation. Pour le cas des approches statiques, certes très rapides mais les tâches de modélisation et d'estimation sont complexes.

3.1.2.1 Les travaux visant les architectures monoprocesseur

Dans ce type d'architecture, il y a une seule unité de contrôle. L'analyse des performances est réalisée conjointement sur les parties logicielles et matérielles. On peut citer les principaux travaux [9] :

- PMOSS : calcule l'accélération due au co-processeur sur la performance globale du système pour la partie matérielle. Il utilise pour cela des analyses statiques (calcule le temps d'exécution en fonction des descriptions de la machine de contrôle).

- PMOSS utilise pour le logiciel, des analyses statiques et dynamiques. Pour les communications des analyses dynamiques sont utilisés.
- COSYMA : calcule des métriques séparées pour le logiciel, le matériel et la communication. Ces métriques serviront par la suite à des équations basée sur le recuit simulé pour effectuer les différentes estimations au niveau matériel, logiciel et communication.
- LYCOS : effectue des estimations de performance en utilisant des analyses dynamiques et mesure le temps d'exécution à bas niveau pour le matériel, le logiciel et la communication.

Malgré leur performance, ces méthodes restent limitée pour les architectures complexes contenant plus d'un processeur.

3.1.2.2 Les travaux visant les architecture multiprocesseurs

Dans ce type d'architecture, il y a plusieurs unités de contrôles. On peut citer les principaux travaux [9] :

- SpecSyn : prend en considération des architectures où le nombre de microprocesseurs et de co-processeurs est quelconque. L'approche utilisée est mixte, et se déroule en deux étapes :
  1. Pre-estimation : réalisée avant la phase d'exploration d'architectures. L'approche utilise une analyse dynamique pour extraire des temps d'exécution pour différents nivaux (processus, bloc de base, communication).
  2. Estimation en ligne : effectuée en phase d'exploration. Elle exploite les résultats obtenus en phase de pré estimation, qui sont à leurs tours utilisés dans des équations complexes pour calculer les performances globales du système.

  L'inconvénient de cette approche est son incapacité à capturer les changements dynamiques du comportement temporel durant la phase d'exploration d'architectures. Par exemple, cette méthode est incapable d'estimer le temps d'attente d'un processus avant son exécution. Une telle défaillance influe négativement sur les résultats de l'estimation.
- VCC : résout le problème de capture de changements dynamiques, en fusionnant une simulation de haut niveau avec des estimations de bas niveau. Cependant l'estimation logicielle manque toujours de précisions en raison de l'utilisation d'un modèle générique de processeur. De plus le niveau d'abstraction du langage de spécification à l'entrée est plus bas.
- Yen et al : ils analysent au niveau système, l'interaction entre les différents processus en donnant le meilleur et le pire délai pour chacun d'entre eux. Ensuite en partant d'un graphe acyclique représentant les dépendances de données entre les processus, ils calculent le temps d'exécution dans le pire cas pour le système entier. La méthode est précise et prend en compte les délais de communications. Mais elle présente l'inconvénient de son utilisation limitée à des applications où il est suffisant de connaître les délais dans le pire des cas. De plus, elle nécessite un graphe acyclique pour représenter le processus.

Généralement, pour l'estimation des performances on prend comme paramètre : le temps d'exécution, la surface et la consommation.

3.1.3 Evaluation de la consommation d'énergie

La consommation d'énergie est devenue critique dans la conception de processeurs embarqués. Les concepteurs doivent expérimenter des différents niveaux du software et d'architecture et évaluer diverses techniques d'optimisation de la consommation d'énergie. Les outils d'évaluation de la consommation d'énergie au niveau architectural deviennent de plus en plus importants avec la complexité croissante de la conception actuelle des SOC [10]. Par exemple, le développement des téléphones de troisième génération capables de délivrer des images et du son, fait augmenter les besoins en énergie, en ressources matériels ainsi qu'en puissance de traitement. Cependant, la technologie de batteries progresse moins vite que la puissance dissipée par les applications, il faut trouver de nouveaux moyens pour étendre la durée d'utilisation.

3.1.3.1 Outils d'estimation de la consommation au niveau processeurs

Les outils actuels de simulation comme PSPICE, POWERMIL [13] permettent de calculer la consommation d'un processeur à partir de son modèle VHDL. Mais d'un point de vue pratique, cette

technique est peu utilisable, on utilise souvent des modèles abstraits. Par exemple, l'expression suivante permet de calculer et estimer la consommation d'un programme :

$$E = \sum_{i}(R_i \times N_i) + \sum_{i,j}(O_{(i,j)} \times N_{(i,j)}) + \sum_{ki}(E_k) \quad (7)$$

Où :

- O(i,j): coût dû aux changements de chemin de données et de contrôle entre deux instructions consécutives (overhead).
- Ri : coût de base d'une instruction en consommation.
- Ni : nombre de cycles nécessaires à l'exécution du programme.
- Ek : représente la consommation due aux cycles perdus, par exemple, lorsque
- les données ne sont pas dans le cache (*cache stall*) ou que le pipeline (*pipeline stall*) doit être vidé.
- N(i,j) :le nombre de fois où les instructions i et j sont exécutées consécutivement.

Parmi les outils qui ont été élaborés pour estimer rapidement la consommation d'un processeur exécutant une application, on cite :

3.1.3.2 Techniques employées pour la réduction de la consommation

Deux méthodes existent pour étendre l'autonomie de fonctionnement du système [10]:
La première augmente la quantité d'énergie embarquée, malgré les nombreuses recherches enregistrées dans le domaine des batteries. Ça reste un grand défi, car on ne peut pas augmenter les capacités d'une batterie sans augmenter le poids, le volume et le prix.

La seconde solution a donné lieu à de nombreuses recherches dans le domaine de l'électronique et de l'informatique.

Les techniques utilisées pour réduire la consommation énergétique sont :

- Les techniques matérielles : se sont les plus répondues, elles se basent sur la conception de composants spécifiques à faible consommation énergétique.
- Les techniques logicielles : consistent à modifier le logiciel pour diminuer le coût énergétique de son exécution (par exemple optimisation du code des applications).
- Les techniques de conception hybrides : consistent à réaliser une collaboration entre le logiciel et le matériel afin d'optimiser la consommation totale.

3.1.4 Evaluation de la surface silicium

Dans le cas de l'optimisation de l'architecture à base de simulation, on peut utiliser les outils suivants pour évaluer la surface [10]:

- L'outil CACTI : estime la surface en silicium du cache instruction et cache données utilisés par le processeur.
- L'outil FUPA : estime la surface en silicium des unités de calcul avec virgule flottante.

3.2 SimpleScalar

SimpleScalar est un outil qui permet d'évaluer les performances des processeurs super scalaires [14].

Les principaux avantages sont sa portabilité (il a en effet été testé sur de nombreuses Platte formes), sa flexibilité vison extensibilité et ses performances. Il est notamment permis de le modifier selon sa convenance pour un usage personnel. Les tests standard (benchmarks) de la SPEC2000 ont été spécialement compilés pour pouvoir être utilisés sur ce simulateur. Mais il est aussi possible de tester ses propres programmes (écrits en C ou en FORTRAN) en les compilant avec une version adaptée de GNU GCC, fournie dans la suite SimpleScalar3. La simulation produit des statistiques détaillées sur l'utilisation des ressources internes (le taux de défauts de cache «Miss-rate », nombre d'accès TLB, nombre de sauts mal prédits, etc.) mais fournit également une trace de l'exécution [10].

3.2.1 Simulation de cache : sim-cache et sim-cheetah

SimpleScalar est distribué avec deux versions gérant les caches : sim-cache et sim-cheetah. Ces deux simulateurs utilisent le fichier cache.c et ils utilisent respectivement les fichiers sim-cache.c et sim-cheetah.

Le simulateur sim-cache permet de configurer la taille des deux niveaux de caches, de TLB (Translation Lockaside Buffer) aussi il est également possible de générer un fichier de statistique.

Le simulateur sim-cache utilise les arguments se trouvant dans le tableau 3.

La configuration du cache (<string>) est définie comme telle :

<name> :<nsets> :<bsize> :<assoc> :<repl>.

Avec:

- <name> : nom du cache crée.
- <nsets> : nombre d'ensembles dans le cache.
- <bsize> : taille d'un bloc dans le cache (pour TLB, utiliser la taille d'une page).
- <assoc> : associativité du cache (doit être une puissance de 2).
- <repl>: politique de remplacement (l = LRU, f = FIFO, r = random).

Pour avoir la taille du cache, il faut donc multiplier : <nsets>, <bsize> et <assoc>. Il est possible d'avoir un niveau unifié en faisant «pointer» le cache d'instruction vers le nom du cache de données correspondant au même niveau.

Par ailleurs, sim-cheetah permet en une seule simulation de tester plusieurs configurations de la mémoire cache.

Argument	Description	Valeur par défaut
cache :dl1 <string>	Cache de données de premier niveau	dl1 :256 :32 :1 :l
cache :dl1 <string>	Cache de données de deuxième niveau	ul2 :1024 :64 :4 :l
cache :dl1 <string>	Cache d'instructions de premier niveau	il1 :256 :32 :1 :l

Tableau 3 : Les arguments du simulateur sim-cache.

4 Conclusion

L'application de cette étude est destinée à d'optimisation de l'allocation de mémoire cache dans les systèmes embarqués. En effet, cette optimisation permet de faciliter la représentation d'objet complexe dans un espace limité tout en gardant la même qualité représentative.

Par la suite, on va appliquer notre approche c'est à dire l'application du recuit simulé pour améliorer la sélection des individus et les algorithmes génétiques multi objectifs pour prendre en considération les deux contraintes de temps d'exécution et de consommation en énergie.

5 Bibliographie

1 .P.Parrend, RTOS et gestion matérielle de la mémoire, 2005.
2. M.Koudil, Méthode de conception conjointe des systèmes embarqués, 2004.
3. P.Kadionik, Les systèmes embarqués : une introduction, Décembre 2005.
4. S.Niar, Rapport d'activité du laboratoire d'automatique et de mécanique industrielles et humaines LAMIH, 2005.
5. F.Simonot-Lion, Y.Song, Design and validation process of in-vehicule embedded electronic system.
6. Réussir la révolution de l'électronique embarquée : Les systèmes mono puces (System-on-Chip, SOC), Forum TIMA, http://www.minatec.com/agenda/journee-soc2004_tima.htm.
7..M.T.Kimour, Approche orientée modèles pour le développement des systèmes temps réel et embarqués, 2005.
8. A.Jerraya, De l'ASIC au SoC au réseau de composants sur puce.
9. A.Baghdadi, Exploration et conception systématique d'architectures multiprocesseur mono puce dédiées à des applications spécifiques, 2002.

10. M.K.Ghali, Méthodologie de conception système à base de plate formes reconfigurables et programmables, Mars 2005.
11. http://www.revueonline.com.
12. Y.Ilas, S.Niar, Présentation du simulateur d'architecture de processeurs : SimpleScalar 3.0, 1998.
13. P.Guitton, Estimation et optimisation de la consommation lors de la conception globale des systèmes autonomes, Octobre 2004.
14. M.ElGuelbi, Adaptation des ressources aux besoins des applications par l'utilisation de méta heuristique dans les systèmes embarqués, 2005.

Vers une approche distribuée pour l'ordonnancement réactif sous incertitudes

S. Ourari[1], C. Briand[2] et B.Bouzouia[1]

[1] CDTA, B.P.17 Baba Hassen, Alger, s_ourari@yahoo.com
[2] LAAS-CNRS, 7 Av Colonel Roche, Toulouse, briand@laas.fr

Résumé: Cet article s'intéresse au problème job shop classique à plusieurs machines, noté $J_n||C_{\max}$. Contrairement aux approches classiques, on suppose que les travaux à réaliser ne sont pas tous connus initialement et qu'ils sont définis au fur et à mesure que les commandes apparaissent, l'ordonnancement étant alors adapté de façon réactive. Le fait que l'approche d'ordonnancement proposée dans cet article soit distribuée constitue une deuxième originalité. Chaque ressource gère son propre ordonnancement local (ordonnancement à une machine), l'ordonnancement global résultant d'une coopération entre les diverses ressources. On suppose que les ordonnancements locaux sont des ordonnancements incorporant de la flexibilité séquentielle, cette flexibilité permettant d'une part à chaque ressource de négocier avec les autres et, d'autre part, de faire face aux incertitudes liées à la mise en œuvre. L'objectif global recherché est la cohérence des décisions prises sur chaque ressource avec maintien d'un niveau de flexibilité donné. Pour cela, un modèle de contraintes est défini et des mécanismes de coopération entre ressources, permettant la négociation de décision d'ordonnancement, sont proposés.

1. Introduction

Dans le contexte actuel de l'évolution de l'entreprise, la fonction d'ordonnancement joue un rôle essentiel dans l'organisation de la production. Un problème d'ordonnancement est caractérisé par un ensemble de tâches, dont l'exécution nécessite un ensemble de ressources partagées. Il s'agit de définir la localisation temporelle des tâches de sorte à respecter, d'une part, les contraintes de précédence liées aux gammes opératoires, d'autre part, les contraintes de ressources (une même ressource ne pouvant réaliser simultanément deux tâches). L'objectif est généralement de minimiser la durée totale d'exécution.
Compte tenu du caractère incertain de l'environnement d'une entreprise, la probabilité qu'un ordonnancement prévisionnel soit exécuté comme prévu est faible [7]. En effet, presque toutes les données déterministes associées à un problème d'ordonnancement sont en réalité sujettes à fluctuation. Parmi les sources d'incertitudes évoquées dans différents travaux [7][13][3], citons : les durées d'exécution des taches variables, les arrivées aléatoires de nouvelles tâches, les modifications de dates de disponibilité et de dates échues, les indisponibilités de ressources.

Pour tenir compte au sein de l'ordonnancement de ces perturbations et pour maîtriser les dégradations de performances, des schémas de résolution sont proposés couplant une partie hors ligne (proactive) et une partie en ligne (réactive). Les notions de flexibilité et de robustesse sont alors définies pour caractériser un système d'ordonnancement capable de faire face aux incertitudes. La flexibilité fait référence à la liberté décisionnelle dont on dispose durant la phase d'exploitation d'un ordonnancement prévisionnel [12]. La notion de robustesse est étroitement liée à celle de flexibilité, en effet, de la flexibilité est souvent injectée dans une solution déterministe afin de la rendre robuste. On trouve dans la littérature plusieurs définitions de la robustesse en ordonnancement, nous reprenons celle donnée par Billaut dans [3] «Un ordonnancement est robuste si sa performance est peu sensible à l'incertitude des données et aux aléas ».

Une classification de référence des méthodes d'ordonnancement avec prise en compte des incertitudes est donnée dans [7][17] et [13]. On distingue les méthodes générant un ordonnancement prévisionnel unique, puis proposant de façon réactive des modifications locales afin de prendre en considération l'état nouveau du système de production et d'absorber les conséquences des aléas [19]. D'autres approches, communément appelées approches proactives, se restreignent à des perturbation spécifiques et proposent des méthodes adaptées [16][20][22]. Une autre classe de méthode introduit dans l'ordonnancement prévisionnel de la flexibilité relative à l'ordre d'exécution des tâches, tout en garantissant une performance, ce qui permet de mettre en évidence une famille d'ordonnancements au lieu d'un ordonnancement unique. Ces méthodes sont qualifiées de robustes dans le sens où elles permettent de réagir en ligne aux événements imprévus qui apparaissent durant l'exécution, en autorisant le passage d'une solution devenue obsolète à une autre, sans remise en cause des calculs déjà effectués. Parmi ces méthodes, on distingue les approches basées sur la notion de groupe d'opérations permutables [2][3][11] qui autorisent la permutation de certaines tâches contiguës sur une ressource. Dans [5], pour le cas du problème à une machine , une façon de déterminer une famille flexible de solutions est proposée qui repose sur l'uti1isation d'un théorème de dominance. C'est cette approche qui est à la source des idées proposées dans cet article.

De façon classique, la résolution en-ligne ou hors-ligne de problèmes ordonnancement avec prise en compte des incertitudes est classiquement assimilée à un problème de décision global car la fonction ordonnancement gère l'organisation de la totalité des ressources, chacune devant respecter la solution établie. Pourtant, dans de nombreux champs d'applications (chaînes logistiques, projets industriels, …), les ressources exécutantes sont souvent réparties au sein d'un ensemble d'acteurs, se trouvant parfois en situation de concurrence, et disposant de fait d'une autonomie de décision dont il est fondamental de tenir compte. Il est alors nécessaire d'adopter une démarche coopérative entre les différents acteurs de manière à converger vers un compromis satisfaisant les exigences de performance globale et celles de performance locale. Dans cet article, l'ordonnancement est considéré comme une fonction distribuée où la solution globale résulte d'une négociation entre plusieurs ressources, appelées centres de décision dans ce qui suit, chacun gérant son propre ordonnancement local.

Cet article est structuré comme suit. Dans un premier temps, le problème global considéré, ainsi que l'approche robuste utilisée pour la détermination d'un ordonnan-

cement local au niveau de chaque centre de décision, sont présentés. Dans la partie 3, nous proposons une structure décisionnelle distribuée et formalisons les différents mécanismes de coopération. Un exemple illustratif est donné. Enfin, des règles de coopération sont proposées afin de trouver le meilleur compromis performance/flexibilité.

2. Problématique

Le champ d'application de l'ordonnancement considéré dans ce papier est celui des systèmes de production automatisés. Le problème job shop à plusieurs machines, noté $J_n||C_{\max}$, consiste à ordonnancer un ensemble T de travaux sur un ensemble M de m machines $M=\{M_1,...M_m\}$. Chaque travail est caractérisé par une gamme opératoire et est constitué d'un ensemble de tâches devant chacune s'exécuter sur une des m machines de M dans l'ordre défini par la gamme. On note que chaque tâche i est caractérisée par une durée opératoire p_i et est réalisée par une et une seule ressource disponible en un exemplaire unique. L'objectif est de minimiser la durée totale d'exécution, appelée makespan et notée $C_{\max}$.
Ce problème est NP-difficile [15]. Il est possible de le décomposer en m sous problèmes à une machine interdépendants, où chaque tâche est caractérisée par une fenêtre d'exécution $[r_i, d_i]$, pour lesquels on cherche à minimiser le retard algébrique (problème noté $1|r_i|L_{\max}$). Les dates de disponibilité r_i et les dates échues d_i des travaux peuvent être déterminées ainsi que précisé dans [1] :

$$r_i = L(0,i) \text{ et } d_i = L(0,n+1) - L(i,n+1) + p_i \quad (1)$$

où $L(i,j)$ désigne la longueur du plus long chemin de i à j sur le graphe conjonctif du problème, longueur qui peut être déterminée grâce à l'algorithme de Bellman-Ford (cf. [10]). On remarque que $L(n+1,0)$ est initialisé à la valeur $(-C_{\max})$, où $C_{\max}$ correspond à la valeur courante du makespan, cette valeur pouvant être progressivement augmentée jusqu'à trouver une solution faisable sur chaque machine ($L_{\max}\leq 0$).

2.1 Une approche d'ordonnancement robuste pour le problème à une machine

Dans les années quatre-vingt, considérant le problème d'ordonnancement à une machine avec fenêtres d'exécution, Erschler et al. [8] mettent en évidence une nouvelle condition de dominance pour la recherche de solutions admissibles. Un ensemble $V=\{1,...,n\}$ de n travaux est considéré, chaque travail $i \in V$ étant caractérisé par une date de disponibilité r_i, une date échue d_i, et une durée opératoire p_i. Un ordonnancement est dit admissible si la machine n'exécute jamais plus de deux travaux au même instant et si les travaux finissent à l'heure, i.e. si la condition (2) est satisfaite. Trouver une solution admissible pour ce problème est également NP-difficile [15].

$$\forall i \in V,\ s_i \geq r_i \ \ et \ \ f_i = s_i + p_i \leq d_i \quad (2)$$

Dans [9]et [4], il est montré que le sous-ensemble de séquences caractérisé par la condition d'Erschler et al. est également dominant vis-à-vis des critères réguliers d'optimisation, T_{max}, plus grand retard vrai et, L_{max}, plus grand retard algébrique.

a. Théorème de dominance [8]

Définition 1 : Une tâche s est dite sommet d'un problème V s'il n'existe pas $i \in T$ tel que $r_i > r_s \wedge d_i < d_s$.

Définition 2 : Une pyramide P_α associée au sommet s_α est le sous ensemble de tâches : $P_\alpha = \{i \in T / r_i < r_{s_\alpha} \wedge d_i > d_{s_\alpha}\} \cup \{s_\alpha\}$

En se basant sur les notions de sommets et de pyramides, Erschler et *al* dans [8] démontrent le théorème suivant, désigné sous le nom de théorème des pyramides.

Théorème: *Un ensemble dominant de séquences peut être constitué par les séquences telles que :*

- *les sommets sont ordonnés dans l'ordre de leur indice ;*
- *avant le premier sommet, seuls sont placés les travaux appartenant à la première pyramide rangés dans l'ordre de leur date de début ou, en cas d'égalité, dans un ordre arbitraire ;*
- *après le dernier sommet, seuls sont placés les travaux appartenant à la dernière pyramide rangés dans l'ordre de leur date de fin ou, en cas d'égalité, dans un ordre arbitraire ;*
- *entre deux sommets s_k et s_{k+1}, sont placés en premier les travaux appartenant à la pyramide P_k et n'appartenant pas à P_{k+1} dans l'ordre de leur date de fin (dans un ordre arbitraire en cas d'égalité), puis les travaux communs aux pyramides P_k et P_{k+1} dans un ordre arbitraire, et enfin les travaux appartenant à la pyramide P_{k+1} et n'appartenant pas à P_k dans l'ordre de leur date de début (dans un ordre arbitraire en cas d'égalité).*

b. Approche d'ordonnancement robuste à une ressource

Dans [5], il est montré comment, étant donné un problème V_k associé à une ressource k donnée, et son ensemble dominant S^{V_k} de séquences, déterminé selon le théorème des pyramides, il possible d'associer à chaque travail i de V_k un intervalle de retard $[L_i^{min}, L_i^{max}]$ qui se calcule en temps polynomial [4] et où, L_i^{min} (resp L_i^{max}) désigne le retard de i parmi toutes les séquence de S^{V_k}. Le retard algébrique optimal L_{max} est alors tel que :

$$\underset{i \in V_k}{Max}\left(L_i^{min}\right) \leq L_{max} \leq \underset{i \in V_k}{Max}\left(L_i^{max}\right)$$

Il est donc possible de juger si un ensemble dominant de séquences est acceptable ou non, relativement à la performance au pire. D'autre part, il est montré dans [5] comment éliminer de l'ensemble dominant les séquences les moins bonnes, de sorte à améliorer les performances au pire. On note que les valeurs de L_i^{min} et L_i^{max} permettent de déduire les valeurs des dates de début au mieux et au pire s_i^{min} et s_i^{max} de chaque tâche : $s_i^{min} = L_i^{min} + d_i - p_i$ et $s_i^{max} = L_i^{max} + d_i - p_i$ (3)

2.2 Exemple

A titre d'illustration, considérons un exemple extrait de [6] comportant quatre travaux et trois machines. Les données relatives aux durées opératoires p_i et aux machines M_j utilisées pour exécuter les tâches sont données dans le tableau 1 et le graphe conjonctif correspondant à l'exemple est représenté sur la figure 1.

travail	tâche	p_i	M_j
1	1	7	1
1	2	10	2
1	3	5	3
2	4	3	2
2	5	15	1
3	6	30	3
3	7	3	1
3	8	2	2
4	9	6	3
4	10	3	1

Tab 1 : Exemple

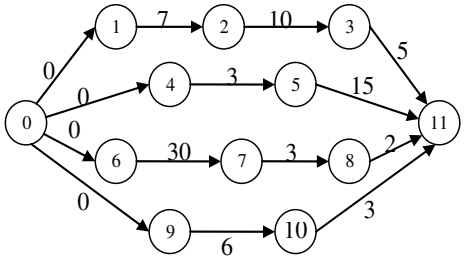


Fig. 1 Graphe conjonctif de l'exemple du tableau 1

Le problème job shop peut être décomposé en 3 sous-problèmes à une machine M_1, M_2 *et* M_3 dont les dates de début au mieux et au pire $[r_i\ d_i]$, calculées par propagation, sont précisées sur la figure 2.

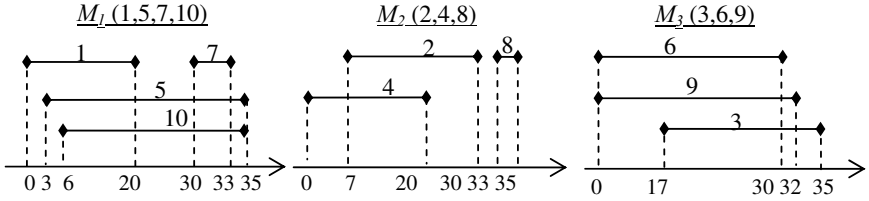


Fig 2. Structures d'intervalles de l'exemple de la figure 1

Pour chaque problème à une machine M_k, l'utilisation du théorème des pyramides permet de déterminer l'ensemble des séquences dominantes S^{Vk} et de déduire, d'après les valeurs de L_i^{min} *et* L_i^{max} (cf. équations (3)), les dates de début au mieux s_i^{min} (resp. au pire s_i^{max}) ainsi que les dates de fin au mieux f_i^{min} (resp. au pire f_i^{max}). L'utilisation de ce théorème sur notre exemple donne les résultats illustrés par le tableau 2. Seule la machine M_1 possède une flexibilité séquentielle puisque la structure d'intervalles qui lui est associée permet de caractériser 4 séquences dominantes.

M_k	M_1	M_2	M_3
S^{Vk}	(1,5,10,7) ;(1,5,7,10) ; (1,7,5,10) ; et (1,10,7,5)	(4,2,8)	(9,6,3)
$[s_i^{min}, f_i^{min}]$; $[s_i^{max}, f_i^{max}]$	i=1 : [0,7] ;[0,7] i=5 : [7,22] ; [36,51] i=7 : [30,33] ; [30,33] i=10 :[22,25] ; [48,51]	i=4 : [0,3] ; [0,3] i=2 : [7, 17] ; [7, 17] i=8 :[33, 35] ; [33, 35]	i=6 : [0,30] ; [0,30] i=9 : [30, 36]; [30, 36] i=3 : [36, 41] ; [36, 41]

Tab 2. Séquences dominantes et dates de début au mieux et au pire.

Nous déduisons donc $41 \leq C_{\max} \leq 51$. On note que l'ordonnancement optimal possède un makespan $C^*_{\max} = 41$.

3. Structure décisionnelle distribuée

Comme nous l'avons mentionné en introduction, la fonction ordonnancement doit être distribuée afin de parvenir à une solution globale résultant d'une négociation entre plusieurs acteurs tout en conciliant les objectifs collectifs et individuels. Ceci justifie notre choix pour une organisation distribuée basée sur le concept de coopération entre centres de décisions [18].

3.1 Organisation coopérative adoptée

Comme évoqué précédemment, nous supposons que chaque ressource est assimilée à un centre de décision, qu'elle gère son propre ordonnancement local et qu'elle dispose donc de sa propre flexibilité décisionnelle. Cette autonomie correspond dans notre cas au nombre de séquences dominantes que le théorème des pyramides permet de caractériser, étant donnée une structure d'intervalles définie par les r_i et d_i des tâches.

Comme l'illustre la figure 3, les centres de décision sont traversés par un flux d'en-cours. Ils s'échangent les données nécessaires à leurs prises de décision. Sur cette figure, le centre de décision *Ressource* reçoit d'un certain nombre de centres de décision amont, qualifiés de *Fournisseurs* les produits qu'il transforme. Une fois le produit transformé, le centre de décision *Ressource* le transfère à son tour vers les centres de décisions aval, qualifiés de *Cliens* chargé de la prochaine transformation de la gamme. On note que les liens physiques (flux d'en-cours) entre les ressources dépendent des gammes opératoires associées aux produits manufacturés ; ces gammes étant supposées initialement connues.

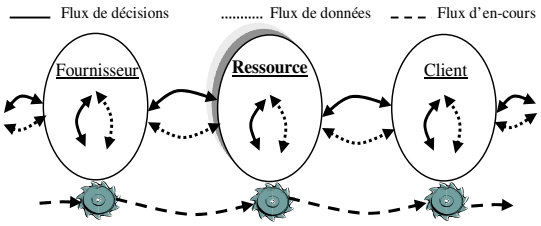


Fig 3. Relations entre les centres de décisions

Les échanges d'information entre centres de décision ont pour but de définir les intervalles de livraisons d'un produit d'un centre vers un autre. Plus précisément, si une gamme opératoire impose que deux tâches i et j soient réalisées en séquence sur les ressources A et B respectivement, alors l'intervalle de début de j sur B, $[s_j^{min}\ s_j^{max}]$, déterminé par construction des séquences dominantes sur B, doit être *cohérent* avec l'intervalle de fin de i sur A, $[f_i^{min}\ f_i^{max}]$, lui-même déterminé par construction des sé-

quences dominantes sur *A.* Nous précisons dans la partie 4 ce que nous entendons par *cohérence.* On peut d'ores et déjà remarquer qu'il est au minimum nécessaire que $s_j^{min} \geq f_i^{min}$ et $s_j^{max} \geq f_i^{max}$ (4).

3.2 Protocole de négociation

Une *négociation* entre une paire de centres de décision *A* et *B* est amorcée lorsque la ressource *B* doit réaliser une tâche *j*, précédée au sens de la gamme par une tâche *i*, réalisée elle même par la ressource *A*. Dans ce cas, le centre de décision *B* émet une *proposition* d'intervalle $[f_i^{min}\ f_i^{max}]$ au centre de décision *A*. Ce dernier peut accepter la proposition ou émettre une *contre-proposition* $[f'_i{}^{min}\ f'_i{}^{max}]$. La conversation s'achève lorsque les deux centres s'entendent sur un intervalle commun. La ressource *A* doit alors respecter autant que faire se peut la valeur $[f_i^{min}\ f_i^{max}]$ contractée avec *B*.

La prise en compte d'une proposition d'intervalle au niveau d'une ressource induit la modification de la structure d'intervalles associée à la ressource et donc la définition d'un nouvel ensemble dominant avec de nouvelles valeurs $[L_i^{min}, L_i^{max}]$ pour chaque tâche. Il s'agit pour la ressource de définir un nouvel ordre entre les r_i et d_i des tâches qu'elle gère de sorte à ce que les valeurs $[L_i^{min}, L_i^{max}]$ soit en cohérence avec les valeurs $[f_i^{min}\ f_i^{max}]$ contractées avec les ressources amont et aval. Dans [21], une aide à la décision est proposée qui permet d'aider un décideur à définir cette structure d'intervalles. Dans le cadre de ce travail, on recherche une façon de définir cette structure qui ne fasse pas intervenir l'homme dans les négociations.

Ainsi que l'illustre la figure 4, l'arrivée dans le système d'une nouvelle commande passée par un client, induit une série de conversations (propositions et contre-propositions) entre les ressources concernées par la réalisation du produit. Ces conversations se propagent de ressource en ressource en remontant la gamme. Lorsque les conversations sont achevées, toutes les tâches nécessaires à la réalisation du produit ont été insérées dans les ordonnancements locaux concernées.

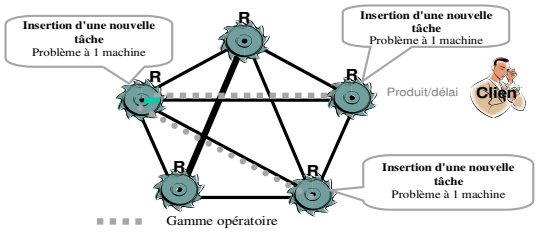


Fig.4 : Prise en charge d'une nouvelle commande

Les ordonnancements locaux sont ensuite adaptés en temps réel (ordonnancement réactif), au fur et à mesure de la réalisation des activités et /ou de l'arrivée de nouveaux ordres de fabrication. Si à un instant donné, du fait de l'occurrence d'un aléa ou

de l'insertion d'une nouvelle tâche, un centre de décision souhaite modifier un intervalle $[f_i^{min}\ f_i^{max}]$ antérieurement contracté avec un centre de décision amont ou aval, il émet une demande de *renégociation*. La renégociation est identique à la négociation (suite de propositions et de contre-propositions) sauf qu'il existe un intervalle $[f_i^{min}\ f_i^{max}]$ servant de référence et qu'un centre de décision recevant une demande de renégociation d'un intervalle a le droit de la rejeter.

Comme on peut à présent le comprendre, la coopération est ici vue comme une façon de parvenir à une solution d'ordonnancement globale en respectant l'autonomie décisionnelle de chaque ressource. Il s'agit de trouver un compromis accordant d'une part à chaque ressource suffisamment de flexibilité pour pouvoir réagir aux perturbations dont elle est victime (pannes, retards, nouvelles activités urgentes, …), sans imposer une remise en cause systématique des décisions d'ordonnancements. D'autre part, le compromis trouvé devra assurer une performance globale satisfaisante relative par exemple à la durée totale de réalisation $C_{\max}$ considéré ici, ou encore au plus grand retard algébrique $L_{\max}$.

3.3 Exemple

On note $\tilde{r}_i$ et $\tilde{d}_i$ les valeurs fixes des dates de début au plus tôt et de fin au plus tard déterminées par propagation de dates sur le graphe conjonctif du problème. Les nouvelles structures d'intervalles (valeurs des r_i et d_i) qu'une ressource définira suite aux négociations avec les ressources amont ou aval, doivent vérifier les contraintes suivantes:

$$r_i \geq \tilde{r}_i \qquad i = 1,...,n_j \text{ et } j = 1,...,m \qquad (5)$$

$$\tilde{d}_i \geq \tilde{r}_i + p_i \qquad i = 1,...,n_j \text{ et } j = 1,...,m \qquad (6)$$

Considérons à nouveau l'exemple donné dans la partie 2. Chaque ressource gère une structure d'intervalles et dispose localement et de ce fait d'un ensemble d'informations sur les dates de début et de fin au pire et au mieux. Ces informations sont communiquées aux ressources qui lui sont liées par une relation de gamme opératoire, comme schématisé sur la figure 4.

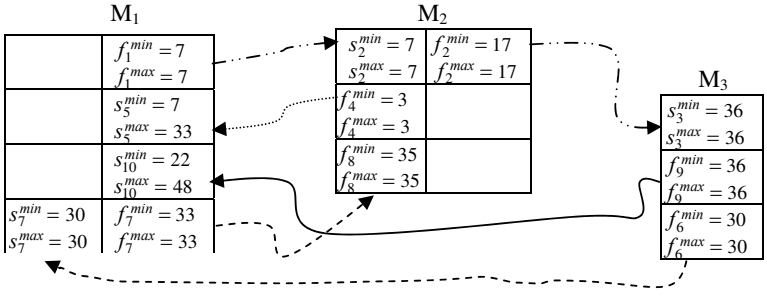


Figure 5. Schématisation des cohérences entre ressources

On se restreint pour cet exemple au seul respect des contraintes minimales de cohérence (4). Il existe une incohérence entre les tâches 9 et 10. Des propositions de négociation sont amorcées par la ressource 1 qui demande à la ressource 3 de diminuer la date de fin de la tâche 9. Comme il n'est pas possible de diminuer la date de début de 9, c'est la ressource 1 qui doit augmenter la date de début du travail 10 en éliminant les séquences dominantes faisant commencer 10 trop tôt. On obtient ainsi les dates au pire et au mieux indiquées sur les figures 6 et 7.

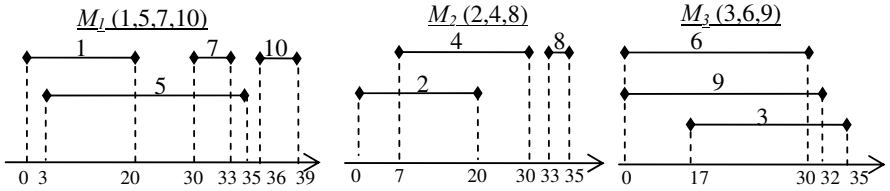


Fig. 6 Nouvelles structures d'intervalles après négociation

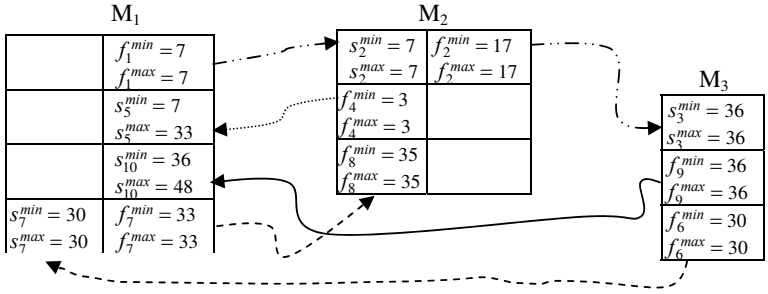


Fig. 7 Schématisation des cohérences entre ressources après négociation

A ce stade, toutes les dates étant cohérentes entre elles, la conversation entre les ressources s'achève, et la solution finale est un ordonnancement dans lequel les ordres de passage sur les ressources 1, 2 et 3 sont respectivement (1,5,7,10), (4,2,8) et (6,9,3), ou bien (1,7,5,10), (4,2,8) et (6,9,3). En ne considérant que les ordonnancement actifs, la ressource 1 peut uniquement retenir la séquence (1,5,7,10) qui conduit à une solution optimale ayant un makespan de 41. Remarquons que la solution optimale donnée dans [6] correspond à l'ordonnancement (1,5,10,7), (4,2,8) et (9,6,3). Voir figure 8.

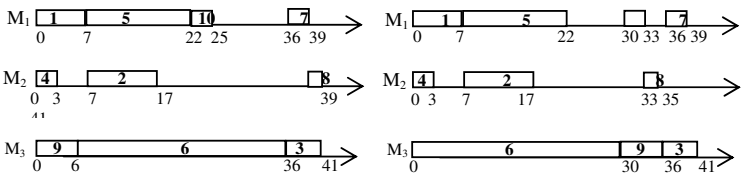


Fig. 8. Solutions optimales

4. Coopération et contraintes

Nous avons dans la section 3 présenté une démarche d'ordonnancement distribué où chaque ressource gère son propre ordonnancement local et veille au respect des contraintes de cohérence par des négociations successives. Cette approche suppose que tous les ordonnancements locaux soient flexibles, cette flexibilité étant la clé permettant l'absorption d'un aléa ou à la prise en compte de nouvelles tâches. Il faut donc éviter la situation où il n'y aurait plus qu'un et un seul ordonnancement valide pour chaque ressource, et aussi celle inverse où les centres de décisions se réserveraient trop de flexibilité au détriment de la performance globale. D'autre part, la coopération suppose aussi que la perte de flexibilité, induite par une remise en cause de décisions d'ordonnancement, soit collectivement répartie entre les centres de décision. Il faut par exemple éviter que ce soit la ressource présentant le moins de flexibilité qui ait à subir la totalité des conséquences d'un aléa (dans la mesure où ces conséquences pourraient être supportées par des ressources ayant plus de flexibilité). Pour garantir ces conditions, nous définissons dans cette partie des règles de coopération supplémentaires, ces règles s'exprimant sous forme de contraintes.

4.1 Contraintes de cohérence

Soient deux tâches *i* et *j*, telles que *i* précède *j*, réalisées par les ressources *A* et *B* respectivement. On s'intéresse aux intervalles $[f_i^{min}\ f_i^{max}]$ et $[s_j^{min}\ s_j^{max}]$ (respectant les contraintes (4)) correspondant respectivement aux valeurs au mieux et au pire de fin de *i* sur la ressource *A*, et de début de *j* sur la ressource *B*.
Ces intervalles peuvent être disjoints. Dans ce cas la date de fin au pire de *i* est compatible avec la date de début au mieux de *j* et le risque d'incohérence est nul, mais, la réalisation de *j* ne pouvant se faire immédiatement après la fin de *i,* il sera nécessaire de stocker le produit ayant subi la tâche *i* avant de pouvoir procéder à la tâche *j* (ce qui induit une perte de performance globale). Les intervalles peuvent éventuellement aussi se recouvrir. Autoriser le chevauchement est intéressant car cela permet de gagner sur les temps d'attente entre ressources et donc d'obtenir une performance globale meilleure. En contrepartie, il existe un risque, en cas d'aléas, pour que la date effective de fin de *i* ne soit pas compatible avec la date de début effective de *j*.

Afin de favoriser la performance globale tout en assurant un risque d'incohérence faible, il est intéressant d'introduire la contrainte suivante, où $\tau_{coh}^{\min}$ et $\tau_{coh}^{\max}$ désignent respectivement des seuils d'incohérence, minimal et maximal que l'on impose.

$$\tau_{coh}^{\min} \le \tau_{coh} = \frac{f_i^{\max} - s_j^{\min}}{s_j^{\max} - f_i^{\min}} \le \tau_{coh}^{\max} \qquad \forall (i,j) \text{ tel que } \ i \prec j \qquad (7)$$

Si, lors des négociations, les centres de décision se communiquent les valeurs courantes de τ_{coh}, il semble logique qu'il revienne au centre de décision présentant le plus fort τ_{coh} de faire l'effort le plus grand pour rendre cohérent une paire d'intervalles.

4.2 Contraintes de flexibilité

La cardinalité de l'ensemble de séquences dominantes [3] définit la flexibilité décisionnelle disponible au niveau de chaque ressource. Dans le but d'assurer globalement un même niveau de flexibilité pour chaque ressource, il est intéressant de contraindre la perte de flexibilité, lorsqu'on passe d'une structure d'intervalle Δ à une autre structure $\tilde{\Delta}$, de sorte que cette perte soit toujours inférieure ou égale à une valeur supérieure $\tau_{flex}^{\max}$.

$$\tau_{flex} = \frac{\left\| S_{dom(k)}^{\tilde{\Delta}} \right\| - \left\| S_{dom(k)}^{\Delta} \right\|}{\left\| S_{dom(k)}^{\Delta} \right\|} \leq \tau_{flex}^{\max} \qquad k = 1,\ldots,m \tag{8}$$

Comme pour τ_{coh}, il est intéressant que les centres de décision se communiquent les valeurs courantes de τ_{flex} le centre de décision présentant le plus fort τ_{flex} doit logiquement faire l'effort le plus important pour rendre les intervalles cohérents.

5. Conclusion

Dans cet article, le problème d'ordonnancement de type job shop en environnement perturbé est considéré. Contrairement aux approches classiques centralisées de résolution, une approche distribuée a été proposée qui accorde à chaque ressource une autonomie décisionnelle. Les décisions d'ordonnancement résultent des négociations et renégociations entre paires de ressources liées par les gammes opératoires. La coopération conduit à la caractérisation d'une famille de solutions permettant de faire face aux aléas. Les négociations et renégociations sont amorcées lors de l'insertion de nouvelles tâches ou à l'occurrence d'un aléa. Elles induisent la redéfinition des structures d'intervalles associées aux ressources. Généralement, plus la flexibilité séquentielle est grande, plus la capacité de réagir à des aléas de forte amplitude est importante, mais plus aussi la performance générale est susceptible de se dégrader. Inversement, moins la flexibilité est grande, plus l'ordonnancement global est performant, mais moins il est sûr. Des contraintes de cohérences et de flexibilité ont été proposées dans le but de trouver le meilleurs compromis flexibilité / performance. Les bases de la coopération étant posées, il est à présent nécessaire de formaliser davantage les processus de négociation afin d'implémenter cette approche en vue de sa validation et de son amélioration.

Références

[1]. J.Adams, E.BAlas & D.Zawack. The shifting bottleneck procedure for job shop scheduling. Management Science, vol.34, n°3, pages 391-401, 1988.

[2]. A. Aloulou & M-C Portmann, Définition d'ordonnancements flexibles. Première application à un problème à une machine. GI'2001, Aix-en-Provence, 2001]

[3]. Billaut J.C., Moukrim A., & Sanlaville E., Flexibilité et robustesse en ordonnancement. Hermes, Traité IC2, ISBN 2-7462-1028-2, 2005.
[4]. C.Briand, M-J.Huguet, H.T.La & P.Lopez. Approche par contraintes pour l'ordonnancement robuste. Dans J-C.Billaut, A.Moukrim & E.Sanlaville, editeurs, Flexibilité et Robustesse en Ordonnancement, Traité IC2, ISBN 2-7462-1028-2, 2005, pp. 191-215.
[5]. C. Briand, H. Trung La, & Jacques Erschler, A robust approach for the single machine scheduling problem. Journal of Scheduling, Special Issue on Project Scheduling under Uncertainty, Demeulemeester, E.L. and Herroelen W.S. (eds)., vol. 10, no. 2, 2007.
[6]. Chu C. & Proth J.M., L'ordonnancement et ses applications. Masson, Paris, France, 1996.
[7]. Davenport A.J. & Beck J.C., A survey of techniques for scheduling with uncertainty. Disponible en ligne sur: http://www.eil.utoronto.ca/profiles/chris/chris.papers.html, 2000.
[8]. J. Erschler, G. Fontan, C. Merce & F. Roubellat, A new dominance concept in scheduling *n* jobs on a single machine with ready times and due dates. Journal of Operation Research, vol.1, pp. 114–127, 1983.
[9]. Erschler J., Fontan G. & Merce C., Un nouveau concept de dominance pour l'ordonnancement de travaux sur une machine. RAIRO Recherche Opérationnelle/Operations Research, Vol. 19, n°1, 1985.
[10]. P.Esquirol & P.Lopez. L'ordonnancement. Economica, 1999.
[11]. C.Esswein, A.Puret, J.Moreira & J.C. Billaut, An efficient methode for job shop scheduling with sequential flexibility. In ORP3, 2003, Germany
[12]. GOThA, Flexibilité et robustesse en ordonnancement. Le bulletin de la ROADEF, 8 :10-12, 2002.
[13]. Herrolelen W. &t Leus R., Robust and reactive project scheduling : A review and classification of procedures. International Journal of Production Research, Vol.42, No.8, 1599-1620, 2004.
[14]. Kouvelis P. & Yu G., Robust discrete optimisation and its applications. Kluwer Academic publisher, 1997.
[15]. Lenstra , J.K., Rinnooy Kan A.II.G & Brucker P., Complexity of machine scheduling problems. Annals of Discrete Mathematics 1, 343-362, 1977.
[16]. Leon V.J., Wu S.D., & Store R.H., Robustness measures and robust scheduling for job shops. IIE Transactions, 26(5) : 32-43, 1994.
[17]. Leus Roel, The generation of stable projects plans: complexity and exact algorithms. Thèse de doctorat, Department of Applied Economics, Katholieke Universiteit Leuven, Belgium, 2003.
[18]. E. Monsarrat, C.Briand & P. Esquirol, Aide à la décision pour la coopération interentreprise. Journal Européen des Systèmes Automatisés, 39/2005.
[19]. Sabuncuoglu I., & de Bayiz M., Analysis of reactive scheduling problem in a job shop environement. European Journal of Operational Research, 126 (3) 567-586, 2000.
[20]. Sevaux M. & Sörensen K., A genetic algorithm for robust schedules in a just-in-time environment. Rapport de recherche, University of valenciennes, LAMIH/SP-2003-1, 2002
[21]. La H.T., Santamaria J.-L. & Briand C., Une aide à la décision pour l'ordonnancement robuste en contexte mono-ressource : un compromis flexibilité/performance. 6ème Congrès de la Société Française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF'05), Tours (France), 14-16 Février 2005, pp 101-116
[22]. Wu S.D, Store R.H., & Chang P.C., One-machine rescheduling heuristics with efficiency and stability as criteria. Computer and Operations Research, 20, 1-14, 1993.

Un modèle de prédiction de la performance des processus basé sur les réseaux Bayésiens

M'hamed HANED, Sid-Ali ADDOUCHE, Abderrahman EL MHAMEDI

MGSI, Modélisation et génie des systèmes industriels
IUT de Montreuil, 140, rue nouvelle France - Montreuil-sous-Bois 93100
{m.haned, addouche, a.elmhamedi}@iut.univ-paris8.fr

Abstract. Le but de ce papier est de présenter un modèle de prédiction qui nous permet d'évaluer analytiquement l'impact d'un changement d'état d'un ou de plusieurs indicateurs de performances d'un processus de production de services ou de biens sur sa performance globale. En d'autres termes, il s'agit d'un modèle d'évaluation de la performance par l'analyse du comportement dynamique du changement. Nous monterons comment nous effectuons cette analyse par le biais d'une formalisation en réseaux Bayésiens (RB). Ce choix vient du fait qu'ils aient pour principal avantage de permettre une formalisation d'un domaine de connaissance sous forme de graphe causal et d'y intégrer une notion d'incertitude à partir d'une approche quantitative. Deux notions indissociables des processus de production au sens large. Nous expliciterons la caractéristique majeure de notre formalisation par le RB qui est la stratification hiérarchique. Nous expliquerons aussi comment nous réduisons l'espace d'apprentissage de sa structure (du RB) aux seules variables d'états de performance qui intéressent l'impact à étudier. Cela montrera le gain en temps et en simplicité de calcul pour mettre en lumière les relations de causes à effets du processus. Evidemment, nous joindrons à ce papier un exemple d'application tiré d'un cas d'école. Les résultats seront commentés et nos solutions seront justifiées.

Mots clefs: Processus, système d'indicateur de performance, plan d'action, prédiction, réseau Bayésien.

1 Introduction

Dans l'entreprise complexe du début du XXI siècle, la diversification des objectifs et les interactions entre les variables et indicateurs de performance associés imposent une vision globale des indicateurs, dans leur conception et leur exploitation. Clivilé [6] voit qu'il est naturel de considérer comme système, l'ensemble des indicateurs, définis suivant de multiples critères, à de multiples niveaux et présentant des interactions [4], [10], [14]. Le système d'indicateur de performance fournit les informations nécessaires au système de pilotage rattaché au système d'indicateurs. Sur la base des informations retournées c-à-d l'état de performance constaté, le centre

de décision[1] enclenche ou pas un plan d'action sur la ou les variables d'action considérées.

Le but de ce papier est de construire un modèle de prédiction qui nous permet de prédire l'impact analytiquement d'un état de performance (changement) sur la performance d'un processus en utilisant les réseaux Bayésiens [2], [3], [11].

En résumé, élaboré un modèle de prédiction qui modélise le comportement dynamique du changement sur la performance.

Dans cette optique de modélisation et d'analyse du comportement d'un changement sur la performance d'un processus, les RB son particulièrement bien adaptés de par leurs potentiels de description et de résolution mathématique. Ils permettent de rassembler et de fusionner les connaissances de diverses natures dans un même modèle (retour d'expérience, expertise, observation, etc..). Grâce à son graphe causal intuitif est compréhensible, le RB est un formalisme convivial permettant de représenter les dépendances entre différentes variables du système étudié et grâce à son système d'inférence, il permet de propager l'influence d'une variable aléatoire sur l'ensemble du système [2], [3], [9], [11], [12]. Deux manières possibles de construire un réseau Bayésien. La première suppose que la structure du réseau a été fixée, et où il faudra estimer les probabilités conditionnelles de chaque nœud du réseau. La seconde, où le but est de trouver le meilleur graphe représentant le problème à résoudre.

La section 2 nous donne une revue de littérature sur l'évaluation de la performance, la problématique laquelle nous traitons. Un Rappel sur la théorie des Réseaux Bayésiens est donnée dans la troisième section. Nous expliquerons dans la section 4 la démarche suivie pour la construction du modèle prédictive en utilisant les réseaux Bayésiens. À la section 5, une application et interprétations des résultats sont données et nous terminerons enfin par une conclusion et quelques perspectives.

2. L'évaluation de la performance et système d'indicateur de performance

La définition que nous adoptons sur l'indicateur de performance, homologuée par l'AFNOR, a été formulé par la commission indicateurs de performance de l'AFGI : «un indicateur de performance est une donnée quantifiée, qui mesure l'efficacité et/ou l'efficience de tout ou partie d'un processus ou système (réel ou simulé) par rapport à une norme, un plan ou un objectif déterminé est accepté dans le cadre d'une stratégie d'entreprise»[1].

D'autres part, pour positionner l'indicateur dans son environnement, il faut lui associer :

- Une variable d'action déterminante,
- Un plan d'action où une action de celui-ci.

[1] Centre de décision : un ensemble de ressources possédant une structure de conduite avec, en particulier, un acteur responsable clairement identifié mobilisé en vue de produire une certaine classe d'objets en respectant au mieux une mission impartie.

Dans ce cadre, le rôle de l'indicateur est aujourd'hui bien définit. Il autorise fondamentalement une boucle de retour dans le processus de décision » appelée boucle de pilotage. Rappelons que le pilotage représente l'ensemble des activités décisionnelles relative à la planification, à la programmation, à la coordination et au contrôle d'un système. Sur la base de l'information retournée, le centre de décision enclenche ou pas une action sur la (les) variable(s) d'action considérée(s). En fonction de la performance constatée, un plan d'action est enclenché sur les variables d'action dont l'effet est ensuite mesuré.

On peut distinguer plusieurs types d'indicateurs [4], [7], [10]. Entre autres, selon le niveau hiérarchique : nous distinguons indicateurs stratégique, indicateurs tactique, indicateurs opérationnel : ces indicateurs sont liés respectivement aux horizons stratégiques, tactiques et opérationnels. Chaque niveau présente un horizon temporel différent aux autres.

Dans cette approche par niveau, un lien de subordination existe entre deux indicateurs de deux niveaux différents exprimé par le biais d'équations déterministe où implicite lorsque la variable essentielle de l'indicateur de niveau inférieur, considéré comme local, est vue comme une variable d'action vis-à-vis de l'indicateur de niveau supérieur, considéré comme global. Les variables d'actions sont identifiées (grâce au modèle cause effet) sous la forme d'un arbre de variables. Au niveau stratégique, ces variables sont les facteurs Clés de Succès[2] et de performance de l'entreprise. Au niveau tactique, on retrouve les facteurs de performance[3] et enfin au niveau opérationnel les facteurs de progrès[4] [5], [6]. Un lien de coordination ou de communication existe aussi entre indicateur de performance, il s'agit d'échanges d'informations concernant les objectifs, actions enclenchées et expressions de performance. Ce qui assure la cohérence horizontale du système d'indicateurs, rappelons que les objectifs d'un même niveau ne sont pas contradictoire entre eux.

D'une part, les objectifs doivent être cohérents avec l'objectif global, ce qui est désigné comme étant la décomposition des objectifs. D'autres part, des opérateurs permettant de synthétiser les expressions de performance délivrées par les indicateurs en une expression agrégée qui est communément désignée comme étant l'agrégation des performances.

La fig. 1.(a) représente un diagramme implicite explicatif de positionnement des indicateurs de performance selon leur niveau hiérarchique. En échangeant la direction des arcs, nous obtenons le modèle cause et effet (fig. 1(b)) qui nous a permet de décomposer l'objectif global en objectifs locaux. Ce deuxième modèle exprime bien qualitativement la dépendance causale entre indicateurs d'un niveau inférieur par rapport aux indicateurs du niveau supérieur. L'indicateur d'un niveau supérieur n'est qu'une agrégation des indicateurs du niveau inférieur.

[2] FCS/FCP ont un impact décisif sur les positions de compétitivité dans un secteur donnée. Ils représentent les enjeux de succès auprès de la clientèle.

[3] FCP représentent des enjeux de performance interne, non perçus par le client.

[4] FP sont des leviers sur lesquels il faut agir pour améliorer la performance des activités critiques, au vu du processus et des facteurs de performance considérés.

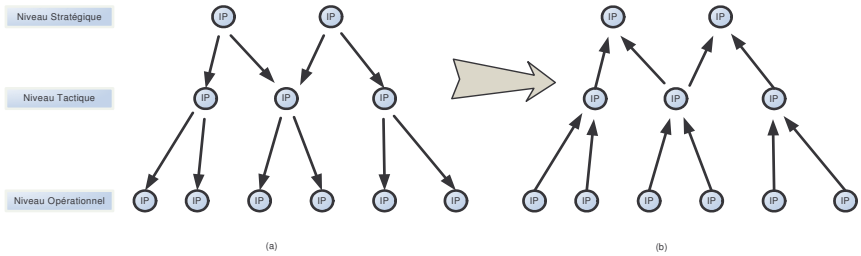


Fig. 1. (a) Système d'indicateurs de performance hiérarchique, (b) Modèle causal associé.

Le rôle du système d'indicateurs consiste à fournir des informations au système de pilotage sur l'état du système piloté. Générer un ou plusieurs plans d'action, ce qui suppose de connaître le détail de l'état espéré sous forme d'objectifs associés aux différentes variables identifiées. Par conséquent, Le pilotage au niveau d'un centre de décision consiste à choisir la meilleure combinaison possible des variables d'action pour atteindre les objectifs espérés.

Pour cela, nous détaillerons dans ce qui suit la construction d'un modèle d'indicateur probabiliste. Le modèle résultant, nous permet de prédire l'état de performance globale du processus par rapport à un plan d'action où un état des variables connues. Les résultats du système d'indicateurs renseignent le centre de décision sur l'état du système piloté.

Hypothèses du travail : nous supposons disposer d'un historique sur les différentes états possible de l'ensemble des indicateurs prisent en considération.

3. Rappel sur la théorie des Réseaux Bayésiens

À la suite des travaux pionniers de Pearl [13], les réseaux Bayésiens ont émergés comme un formalisme puissant qui unifie différents concepts de modélisation probabiliste utilisés en statistique, intelligence artificielle, traitement du signal, etc.

$B = (\xi, \Theta)$ est un réseau Bayésien si $\xi = (X, E)$ est un graphe acyclique dirigé (DAG) dont les sommets représentent un ensemble de variables aléatoires $X = \{X_1, ..., X_n\}$, et si $\Theta_i = \lfloor p(X_i / X_{Pa(Xi)}) \rfloor$ est la table de probabilités conditionnelles (TPC) du nœud i connaissant l'état de ses parents $Pa(X_i)$ dans ξ. Le réseau Bayésien B représente donc une distribution de probabilité sur X qui admet la loi jointe suivante :

$$p(X_1, X_2, ..., X_n) = \prod_{i=1}^{n} P\left(X_i / X_{Pa(X_i)} \right) \quad (1)$$

Cette décomposition de la loi jointe est à l'origine d'algorithmes d'inférence puissants qui font des réseaux Bayésiens des outils de modélisation et de raisonnement très pratiques lorsque les situations sont inceratines ou les données sont incomplètes.

La construction où l'apprentissage d'un réseau Bayésien peut se faire de deux manières différentes selon ce qu'on met à notre disposition :

- L'apprentissage des paramètres, où nous supposerons que la structure du réseau a été fixée, et où il faudra estimer les probabilités conditionnelles de chaque nœud du réseau.
- L'apprentissage de la structure, où le but est de trouver le meilleur graphe représentant au mieux la tâche à résoudre.

4. Construction d'un modèle probabiliste

Nous allons présenter ici les différentes étapes de la construction du réseau Bayésien. Après avoir définis les variables utilisées, nous allons expliquer comment nous avons obtenu le graphe causal. Nous montrerons ensuite comment nous avons défini les états de chaque variable et comment renseigner les différentes tables de probabilités conditionnelles. Finalement nous décrirons, l'utilisation du modèle causal obtenu.

Les indicateurs de performance de chaque niveau fournissent les informations nécessaires pour définir la structure du modèle RB. L'étape suivante consiste en la construction d'un modèle probabiliste qui représente les rapports causaux existants entre les indicateurs de performance.

4.1. Nœuds et états

Les nœuds du RB représentent les indicateurs de performance associés à chaque niveau temporel. Il y'a autant de nœuds qu'il y'a d'indicateurs de performance.

L'essentiel des algorithmes développés pour l'inférence et l'apprentissage dans les réseaux Bayésiens, aussi bien que les outils disponibles sur le marché pour mettre en œuvre ces algorithmes utilisent des variables discrètes. La plupart des variables statiques représentées dans ce réseau sont de nature continue. Toutefois, comme l'exploitation des RB continues est difficile, les variables statiques doivent être discrétisées.

La méthode de discrétisation utilisée consiste à découper l'espace continu de la variable Xj en classes de distance équivalente [12]. Pour ce faire, il est nécessaire de déterminer l'espace d'état continu de la variable et le pas de discrétisation. Le pas de discrétisation d^{X_j} est calculé afin d'obtenir des classes de valeurs X_j d'amplitude

équivalente en choisissant un nombre de classe fixe. Ici, nous supposons avoir trois classes d'états pour chaque variable.

Pour les nœuds racines, nous divisons l'espace d'état sur trois afin d'obtenir trois classes d'états de distance égale. Pour les nœuds non racine, l'espace d'état est déterminé par combinaison des espaces d'état de ces parents, en utilisant la relation de causalité exprimé par le biais d'une équation déterministe (si celles-ci sont identifiées).

L'espace d'état X^{X_j} est alors constitué de trois classes d'intervalles :

$$X^{X_j} = \{[\, X_j \min ,\ X_j \min + d\ [,\ [\, X_j \min + d\ ,\ X_j \max - d\ [\text{ et } [\, X_j \max - d\ ,\ X_j \max\ ]\}. \quad (2)$$

La discrétisation des nœuds qui représentent les indicateurs de performance permet de définir les différents niveaux (classes) de performance que le plan d'action (changement) peut atteindre où provoquer.

4.2. Un arc

L'arc est synonyme de dépendance conditionnelle et causale entre variables qu'il relie. Dans le cadre de ce travail, nous pouvons distinguer deux types de lien :

- Lien entre certains indicateurs de performance exprimée par le biais d'une équation déterministe.
- Les performances d'un niveau son agrégat des indicateurs d'un niveau supérieur par le biais de relations non explicité.

Nous pouvons distingué trois niveaux sur la structure du réseau Bayésien : Le premier niveau 1 correspond aux indicateurs de performance du niveau opérationnelle, le niveau 2 correspond aux indicateurs de performance qui sont agrégation des indicateurs du niveau 1, en l'occurrence niveau tactique. Le niveau 3 correspond aux objectifs globaux exprimés sous forme d'indicateurs du niveau stratégique qui sont aussi résultante d'agrégation des indicateurs du niveau tactique.

Le reste à faire est d'identifier les liaisons non explicite entre les indicateurs. Pour cela deux solutions sont proposées :

- La première, Il s'agit de répondre à des questions de genre Qui influence sur Qui ? Afin d'identifier toutes les influences existantes entre les indicateurs de performance. Cette étape d'identification des relations s'effectue par brainstorming [3], [11] d'experts, c'est-à-dire d'individus possédant une expérience où une connaissance du fonctionnement de l'organisation où d'une partie de processus.
- La deuxième, il s'agit d'effectuer à partir d'un échantillon disponible un apprentissage de structure [9], [11] tout en imposant des contraintes sur l'orientation des arcs. Nous interdisions dans l'algorithme d'apprentissage [3],[9],[12],[14] tout arcs reliant un niveau inférieur avec un niveau supérieur afin de respecter les relations de cause à effet d'un niveau vers un autre. L'apprentissage de structure à partir d'un

echantillon consiste à trouver le modèle réseau Bayésien qui représente le plus fidèlement le problème traité.

Les indicateurs du niveau 3 sont les réponses (états de performance) que doit fournir le système d'indicateur sur l'impact d'un plan d'action sur la performance global du processus.

Nous avons choisit de travailler avec la deuxième solution qui consiste à trouver la structure qui représentera mieux notre problème. On peut énumérer plusieurs algorithme d'apprentissage de structure [9], [11], sur lesquels on peut facilement insérés les contraintes ci-dessus, ce qui réduira la complexité du calcul et facilitera la définition de la structure appropriée.

Le réseau Bayésien résultant (Fig. 2.) est un réseau très simple, qui est composé de trois niveaux, il se caractérise par déconnection des nœuds des niveaux 1 et 2, et connections entre nœuds quand on passe d'un niveau vers le suivant. La majorité des liaisons entre variables (indicateurs) du réseau Bayésien sont construites à partir des connaissances expertes. Le reste des liaisons sont construites en utilisant l'apprentissage de structure à partir d'un échantillon tout en prenant ce qui déjà construit comme initialisation du réseau Bayésien final, cela réduira considérablement la construction de la structure du réseau Bayésien [8].

Les nœuds du niveau 1 représentent les indicateurs de facteurs clés du succès. Dans le cas d'une vision d'une performance interne, se sont des indicateurs clés de performance [6]. Les nœuds du niveau 2 représentent les indicateurs de facteurs de performances et les derniers représentent les indicateurs facteurs clés du progrès.

Grâce au système d'inférence qui caractérise le réseau Bayésien, il permet de propager l'influence d'une variable aléatoire sur l'ensemble du système. Propager les influences cela veut dire échanger les informations entre variables, entre d'autres termes cela permettra la communication entre les indicateur du même niveau afin d'assurer la cohérence horizontale.

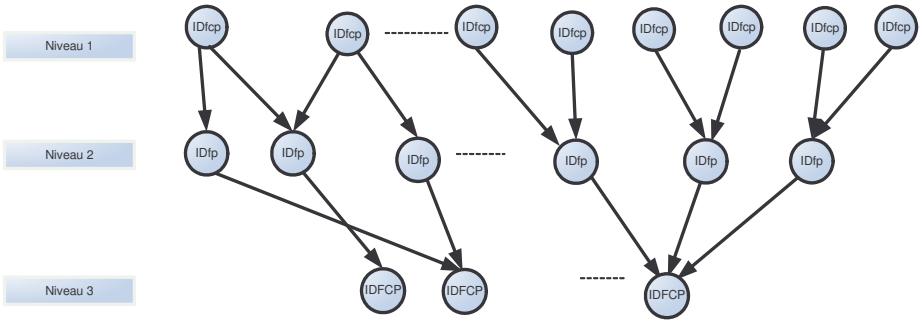


Fig. 2. Réseau Bayésien modélisant le problème.

IDFCS/FCP : Indicateur de facteur clés du succès/ performance.
IDfp :Indicateur facteur de performance.
IDfcp : Indicateur facteurs clés de progrès.

4.3. La table de probabilité conditionnelle (TPC)

Cette étape de construction de notre modèle Bayésien est cruciale, elle permet de quantifier les différentes liaisons existantes entre indicateurs de performance en terme d'influence par le biais des tables de probabilités conditionnelles (TPC). Soit pour les liaisons déterminé par le biais d'équations déterministe où celle déterminé par apprentissage. Pour y parvenir, deux solutions se présentent :

- Réaliser un apprentissage sur un jeu de données [3], [12].
- Faire appel à des connaissances expertes [3], [12].

Le choix de la solution dépend des connaissances disponibles sur le fonctionnement du modèle.

Associer à chaque nœud, la table de probabilité conditionnelle (TPC) quantifie les relations causales. Les nœuds racine ont seulement des distributions de probabilités a priori modélisant la statique de chaque indicateur de performance (niveau 1). Pour le niveau 2 et le niveau 3, une probabilité conditionnelle est affectée pour chaque nœud en considérant chaque configuration de ces parents.

Pour le reste des liaisons apprises par apprentissage, les probabilités conditionnelles sont calculées par l'apprentissage de structure effectué sur cette partie.

4.3.4 Exemple illustratif

Supposons une partie d'un réseau Bayésien, constituée de quatre nœuds représentant les indicateurs de performance. Prenons le cas des indicateurs suivants: ID1, ID2, ID3 et ID4 expriment respectivement des coûts associés. Nous supposons que l'indicateur ID4 s'exprime en fonction des indicateurs ID1, ID2 et ID3 par le biais d'une équation déterministe (coût total d'exécution des activités en série) ci dessous :

$$ID4 = ID1 + ID2 + ID3 \quad (\mathbf{3})$$

Le Tableau 1. ci-dessous représente un échantillon de l'historique associé aux indicateurs cités ci-dessus.

ID1	ID2	ID3
1	2	2
0	1	2
5	5	4
0	4	3
2	5	1
5	0	2

Tableau 1. Echantillon d'un historique sur les indicateurs ID1, ID2 et ID3

L'espace d'état pour ID1 est [0, 5], pour ID2 est [0, 5] et pour ID3 est [1, 4]

L'espace d'état de ID4 est obtenu en appliquant l'équation déterministe (3) pour déterminer la valeur maximum et minimum de son espace d'état qui est [1, 14].

Après discrétisation de ces espaces en trois classes d'état de même distance, nous obtenons selon la formule (2):

Pour ID1 : [0, 1.67 [, [1.67, 3.33[et [3.33, 5] correspondant aux classes de performance 1, 2 et 3 respectivement.

Pour ID2 : [0, 1.67[, [1.67, 3.33[et [3.33, 5] correspondant aux classes de performance 4 , 5 et 6 respectivement.

Pour ID3 : [1, 2[, [2, 3[et [3,4] correspondant aux classes de performance 7, 8 et 9 respectivement.

	ID1	1									2									3								
	ID2	4			5			6			4			5			6			4			5			6		
	ID3	**7**	**8**	**9**	**7**	**8**	**9**	**7**	**8**	**9**	**7**	**8**	**9**	**7**	**8**	**9**	**7**	**8**	**9**	**7**	**8**	**9**	**7**	**8**	**9**	**7**	**8**	**9**
ID4	**10**	49	40.76	43	14	0	12	22	0	1	13	49	8	17	**22**	15	0	0	0	13	39	4	74	63		25	63	50
	11	40	40.43	20	16	15	45	88	0	0	27	3	91	42	50	60	50	1	0	27	22	53	18	37		18	35	24
	12	11	18.80	37	70	85	43	0	1	0	60	48	1	41	28	25	50	0	1	60	39	43	8	0		57	2	26

Table 2. Table de probabilité conditionnelle associée au noeud ID4.

Nous concluons pour ID4 : [1, 5.33[, [5.33, 9.66[et [9.66, 14] correspondant aux classes de performance 10, 11 et 12.

La dernière étape consiste à renseigner les tables des probabilités associées à chaque nœud qui quantifient les relations causales, et les probabilités marginales pour les nœuds racines qui quantifies le statique de la variable. Le tableau ci-dessus (Tableau 2.) représente la table de probabilité conditionnelle associé au nœud ID4 qui quantifie le rapport causal par rapport à ces nœuds parents (ID1, ID2, ID3).

La de la ligne 4 et la colonne 15 (valeur en gras), exprime la probabilité conditionnelle que la variable ID4 soit à l'état 10 sachant que ces nœuds parents ID1, ID2, ID3 sont dans les état 2, 5 et 8 respectivement. Sa forme mathématique est :

$$P(ID4 = 10 / ID1 = 2, ID2 = 5, ID3 = 8) = 0{,}22 \quad (4)$$

Le réseau Bayésien associé est représenté sur la figure ci-dessous

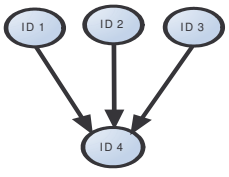


Fig. 3. Réseau Bayésien associé à l'exemple.

4.5 Utilisation

Une fois le graphe est construit, son utilisation repose sur l'algorithme d'inférence qui calcule la probabilité de chaque variable à partir d'un certain nombre de valeurs fixées a priori. Les probabilités de chaque variable (indicateur) permettent de nous renseigner sur leurs états les plus probables, entre autres quantifier les performances associées.

Sur ce modèle nous pouvons tester plusieurs situations de performance et en déduire les causes les plus probables d'occurrence c-à-d faire du diagnostic. Il permet aussi de faire de la prédiction, par à rapport à des observations (par exemple : fixation des variables représentants les deux premiers niveaux), le modèle nous renseignera sur l'état de performance global du processus la plus probable.

5. Application

Dans ce qui suit, nous présenterons un exemple afin d'illustrer l'utilisation possible de notre modèle.

Au départ, nous supposons disposer d'un échantillon d'indicateurs de performance d'un processus donnée. Nous disposons de quatre indicateurs de facteurs clés de progrès du niveau opérationnel, quatre indicateurs exprimant les facteurs de performance au niveau tactique et trois facteurs clés de la performance au niveau stratégique qui expriment la performance globale du processus. Les nœuds représentent les différents indicateurs de performance, organisé selon leur appartenance au niveau temporel (opérationnel, tactique, stratégiques).

Après détermination de l'espace d'état de chaque variable, les différentes relations existantes entre les nœuds d'un niveau à un autre exprimées soit par le biais d'équations où déterminé par apprentissage et renseignement des différentes tables de probabilités, le réseau Bayésien ci-dessus (fig. 4.) représente notre modèle limité à onze nœuds puisque l'outil logiciel BayesLiab [2] utilisée est une version d'évaluation.

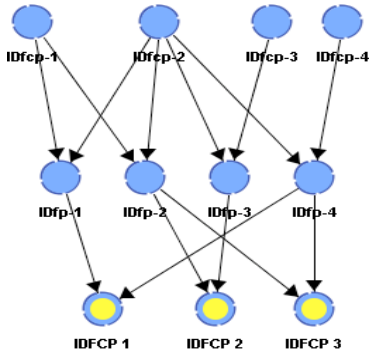


Fig. 4. Le réseau Bayésien modélisant le problème.

5.1 Interprétation de quelques résultats

La figure (Fig. 5.) montre Les probabilités associées à chaque variable permettent de nous renseigner sur leurs états les plus probables, entre autres quantifier les performances associées.

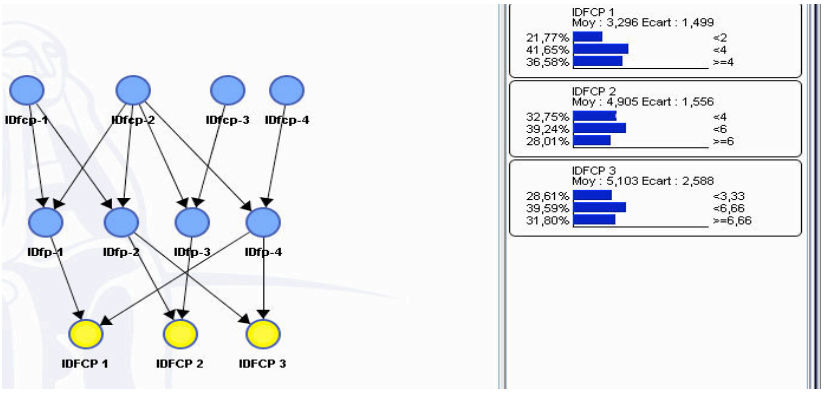


Fig. 5. Etat de performance globale constaté.

L'exemple ci-dessous (Fig. 6.), montre l'état de performance obtenus en fixant les probabilités des variables IDfcp3, IDfp2 aux états >=5 et <0.67 respectivement. L'impact de ce changement est exprimé sur les variables IDFCP 2, IDFCP 3 par les probabilités d'appartenance à chaque classe de performance.

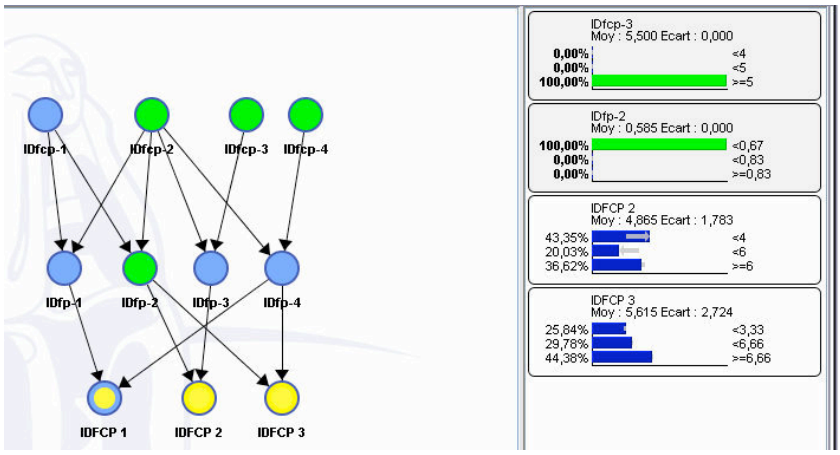


Fig. 6. Nouvelle performance obtenue des indicateurs après changement.

6. Conclusion et perspective

Le Modèle réseau Bayésien nous fournis un modèle de prédiction et de quantification des impacts en relation avec la performance d'un processus provoqué par les changements effectués sur la performance.

Notre réseau Bayésien est structuré en trois niveaux, Nous avons construit la structure associée en deux étapes, la première consiste à l'identification de l'ensemble des variables qui caractérisent le modèle. Ensuite identifier tout les arcs à partir des équations déterministes qui relient certaines variables (entre indicateur de performance d'un niveau avec les indicateurs du niveau qui suit directement), une première construction est réalisée. Le reste des liaisons est déterminée par apprentissage de la structure de réseau Bayésien tout en prenant en compte que les arcs orientés seulement d'un niveau supérieur vers le niveau inférieur qui le suit directement.

La discrétisation des variables associées aux différents niveaux permet de définir des classes de performance que le changement peut atteindre.

En supposons disposer d'un historique, une étape d'apprentissage des paramètres a été effectuée afin de renseigner les tables de probabilités conditionnelles. Le reste des tables de probabilités conditionnelles sont renseignées lors de l'apprentissage de la structure associée.

Le modèle Bayésien construit permet de prédire l'état de performance du processus par rapport à des changements effectués tout en observant les nœuds associés aux indicateurs de performance. Ces informations fournies permettent d'engager les plans d'actions appropriés.

A court terme, en utilisant un historique identique, essayer de construire tout le réseau Bayésien par apprentissage en utilisant la notion de score. Cela nous permettra d'identifier toutes les relations existantes entre les indicateurs. Utiliser le même historique et comparer les résultats obtenus par les deux modes de construction d'un réseau Bayésien.

Références

1. AFGI : Association française de Gestion Industrielle, Evaluer pour évoluer, les indicateurs de performance au service du pilotage. ouvrage collectif AFGI (1992).
2. Bayesia SA, www.bayesia.com/FR/produits/blab/BlabPresentation.php. 16/02/2007.
3. Becker A., Naim P. : Les Réseaux Bayésiens. Edition Eyrolles (1999).
4. Berrah L. : Une approche d'évaluation de la performance industrielle, Modèle d'indicateurs et techniques floues pour un pilotage réactif. Thèse de doctorat en Génie Industriel, INP Grenoble (1997).
5. Clivilé V. : Spécification et modélisation d'un système d'indicateurs de performance. http://www.ifma.fr/AMOEP/GDR-Clivile.pdf, 16/02/07.
6. Clivilé V. : Spécification et modélisation d'un système d'indicateurs de performance. Université de Savoie, Thèse de doctorat en génie industriel (2004), pp 215.
7. Courtois A., & al. : Gestion de Production. Paris, Editions d'organisation (2004).
8. François O., Leray P. : Etude comparative d'algorithme d'apprentissage de structure dans les réseaux Bayésiens. RJCIA 2003, Rencontres Jeunes Chercheurs en IA, Laval (2003).

9. Leray P. : Réseaux Bayésiens : Apprentissage et Modélisation de systèmes complexes. rapport d'Habilitation a diriger les recherches, (2006), pp 199.
10. Lorino P. : Méthodes et Pratiques de la Performance. Paris, Edition d'organisation (2003).
11. Meganck S., Leray P., Maes S., Manderick B.: Apprentissage des réseaux Bayésiens causaux à partir des données d'observation et d'expérimentation. Journée Francophone des Réseaux Bayésiens (2006).
12. Muller A. : Contribution à la maintenance prévisionnelle des systèmes de production par la formalisation d'un processus de pronostic. Thèse de doctorat, Université Henri Poincaré, Nancy I (Nancy) (2005), pp 179.
13. Pearl J, : Probabilistic reasoning in intelligent system : network of plausible inference. Morgan Kaufmann Publishers, San Mateo, CA (1988).
14. Naim P. & al. : Les Réseaux Bayésiens. Edition Eyrolles (2004).
15. Tahon C. : Evaluation des Performances des Systèmes de Production. Paris, HERMES LAVOISIER, IC2 Productique (2003).

Medical Image Segmentation using Enhanced Hoshen-Kopelman algorithm

K. Kies[1], N. Benamrane[1]

[1] Département d'Informatique, S.I.M.P.A, U.S.T.O BP1505 El Mnaouer Oran 31000 (Algeria)
e-mail : kieskarima@hotmail.com

Abstract. We present a method of 2D and 3D segmentation based on the Enhanced Hoshen-Kopelman algorithm and its extension to non-lattice structure. The main feature of this method is to combine a merging strategy of a region growing algorithm with the multiple labeling technique of the EHK algorithm for regular and non-regular lattice. An efficient reconstruction algorithm is then applied to the set of edge points for the obtained segmented regions. The latest uses 3D Delaunay weighted triangulation. The combination of these already known algorithms makes the proposed approach very fast, efficient and appropriate to volumetric segmentation and particularly the anatomical structures in MRI and CTScan images.

Keywords: Medical Images, 2D and 3D Segmentation, region growing, Hoshen-Kopelman algorithm, Delaunay triangulation.

1 Introduction

Segmentation is an indispensable tool in medical image processing and is useful in many applications such as early pathology detection, precise tumor localization, simulation and planning of chirurgical interventions, etc. The principal aim of segmentation is the extraction of significant objects present in the image either by partitioning this image into connected semantic regions or by extracting one or many specific objects from this image.

Proposed approaches attempting to solve the segmentation problem are numerous and extremely varied. The choice of a method depends on the nature of the image, what primitives are to be extracted and what are the usage constraints (algorithm complexity, real time requirement, memory available, etc...). This variety of possible methods shows that the segmentation problem is far from being solved and there exists no single technique satisfying all the image types.

In the literature, two major classes of segmentation algorithms: contour approach based on the estimation of the locale discontinuities of the gray level function, and the region approach based on homogeneous region extraction. Among the later, a fundamental algorithm is region growing which has received considerable attention from 2D an 3D image analysts and researchers. The principle of the technique is to define some criteria, called "homogeneity criteria", which allow regrouping

connected image pixels into larger regions. These criteria are often based on differential surface characteristics of the 3D object and a stable segmentation in quadric pieces is achieved [1]. They can also be obtained from some evaluation function [2] or by RMS-approximation of image points by simple polynomial surfaces (plane or quadric) [3]. Region growing method has also been used for 3D medical image reconstruction where the voxel gray level defines some "density" of the volume to reconstruct as well as a regrouping criterion.

Another approach that is fundamentally geometrical: mathematical morphology, which mainly consists on comparing objects under analysis to another object with known shape. Morphologic operators applied to 3D images are described in [4] where temporal image sequences considered as 3D images are segmented. Three-dimensional segmentation, based on mathematical morphology, has been applied to anatomical structures in MR images on large databases [5].

In the last two decades, a promising mathematical framework based on variational models and partial differential equations has been developed and used to solve the segmentation problem. There exist two major categories of models: parametric models introduced by Kass et al [6] and geometric models based on curve evolution theory and level sets [7] and [8]. They allow segmentation and object reconstruction by providing compact geometric representations where evolutions laws similar to physical mechanics laws are introduced (forces, physical constraints and friction) [9-14]. Procedures using deformable models have been widely used for segmentation since they are well suited for object extraction from temporal and volumetric data from image sequences such as IRM, CT scanned, X Tomography, ARM and Echography [13-15]. These procedures have been more efficient by using 3D Pyramid images [16] and such hierarchical structures allow "coarse to fine" type segmentation. Region and edge information has been integrated into these models making them more efficient [17].

In some applications, the segmentation is viewed as pattern recognition problem. Thus, the task of delimiting regions is associated to a high level ability to classify regions in particular semantic classes. These approaches are widely used for segmenting cerebral MRI; a review is given in [18]. Among these, we can cite : statistical methods using Markov fields and Bayesian classification [19], the non parametric methods such K-NN algorithms introduced by Duda in 1973, [20], and Parzen windows, fuzzy logic based methods [21], and the so called "Biologists methods" (neural networks, genetic algorithms, etc.)[22-24].

In quest of better efficiency and precision various hybridations of different techniques have been successfully attempted [25-30].

In this paper, a method for 2D and 3D Image segmentation based on the Hoshen-Kopelman algorithm is proposed. The main feature of this method is to combine a merging strategy of a region growing algorithm with the multiple labeling technique of the EHK algorithm for regular and non-regular lattice. This procedure is very efficient and fast and may be adapted to the segmentation of anatomical structures in MRI and CTScan images which are in general quite large. An efficient reconstruction algorithm is then applied to the set of edge points for the obtained segmented regions. The latest uses 3D Delaunay weighted triangulation. The later is very efficient and fast approach makes it possible to adapt it to the segmentation of anatomical structures in MRI and CTScan images.

2 The Hoshen-Kopelman Algorithm

Connected component analysis of grey level image divides this image into regions by labeling uniquely each pixel to a defined region. The regions discriminated by the connected component depend on the definition of neighborhood. Several definitions of neighborhood exist, however the most commonly used is the 4-connected. Basically, the algorithms for connected components assign new labels to the first pixel of each component and attempt to propagate these labels through the neighbors. In image processing, Rosenfeld et al, [31], described an algorithm based on the classical component algorithm for graphs which is known as the classical algorithm for connected component analysis. The algorithm makes two passes through the image. The first pass assigns and propagates labels, while at the same time it creates an equivalence class table. The second pass re-assigns the labels that belong to the same class using the equivalence class table, such that all equivalence is assigned with a unique label. The final image has its connected components indexed with a unique label. While being relatively fast this algorithm requires a large amount of memory to store the equivalent class table, which depends on the image size and on the complexity of the images. The most recent algorithms, for connected component analysis, attempt to overcome the fundamental problems of either being too slow or requiring a large amount of computer memory.

In physics connected component analysis is known as cluster analysis and it is widely used for problems related to the percolation phenomena [32]. In mid 1970s, J. Hoshen and R. Kopelman published an ingenious and efficient cluster analysis algorithm, which is known as the Hoshen-Kopelman (HK) algorithm. This algorithm is linear in time and utilizes small computer memory. It remains an important breakthrough in statistical physics, since it allows the study of cluster size and the percolation problem through computer simulations. Recently, Hoshen et al. developed the Enhanced Hoshen-Kopelman (EHK) algorithm [33], which is an improvement toward cluster analysis. It is capable to yield various shape parameters and still preserves the computational time and space complexities of the original HK algorithm. The key idea for the efficiency of the HK algorithm, which is also essential in the EHK algorithm, lies on the concept of multiple labeling technique. The multiple labeling technique assigns different labels to pixels belonging to the same cluster. Unlike the classical connected component algorithm, which stores an equivalence class table and the equivalence are executed by a second pass over the image. The HK algorithm generates the proper label by assigning negative indexes to equivalent labels and recursively searching for the final label.

In the statistical physics terminology, an image is referred to as a lattice, the connected components are called clusters and a pixel is known as a site. The algorithm begins by scanning the lattice, pixel by pixel or site by site, from up to down and right to left direction (and back to front for 3D images).

3 Segmentation

Grey level segmentation may be achieved using standard region growing technique which is a merging process at various levels. Usually, two or more steps are used; the first step (step 1) operates at the pixel level which is referred as a pre-segmentation step. At this stage the lattice is regular. The further segmentation merging steps consiste on whether regions obtained from step 1 may be merged or not. The first two segmentation steps may make use of uniformity predicates associated to the characteristic attributes of region say Ri are as follow:

1. $P_1(Ri) = [(MAXi- MINi) < Threshold1]$ where MAXi and MINi are respectively the minimum and maximum gray level in region Ri

2. $P_2(Ri) = [V(Ri) < Threshold_2]$ where V(Ri) is the gray level variance of points in region Ri . We have $Threshold_2 = (Threshold\ 1)^2$

The thresholds may be chosen in any standard manner, this issue is not discussed here. Other merging steps may make use of other predicates such as shape or texture.

In this work, we propose an extension to the EHK algorithm for grey level image segmentation. While the main features of the algorithm are preserved, the notion of non occupied site does no longer apply here since all sites (or pixels) are labeled. The background is also a region. For step 1, the standard HK algorithm is used since at the pixel level, the lattice is regular. The clusters obtained correspond to the regions of the pre-segmentation step. After the completion of step 1, a region adjacency graph may be constructed. This graph corresponds to a non regular lattice. This latter may become quite large for very large images. In order to perform step 2 and further merging steps in an efficient manner, we propose and make use of the Extension of Hoshen-kopelman algorithm to non-lattice environments [34].

Relative to other segmentation methods which are at best *n log n* in computational time (*n* being, the number of pixels or voxels), this method is linear in *n*. On Fig. 1, we have plotted the computational time in seconds for random binary 3D images ranging from 100^3 to 300^3 (a 2.6 GHz Intel Pentium IV with 2 GB DDR400 RAM was used).

Two more features make this method very attractive. Firstly, the memory requirement depends solely on the maximum number of labels needed and not on the size of the image. Secondly, this method can be highly parallelized as shown in Tiggemann work on percolation [36] where system sizes as high as 20000^3 where considered.

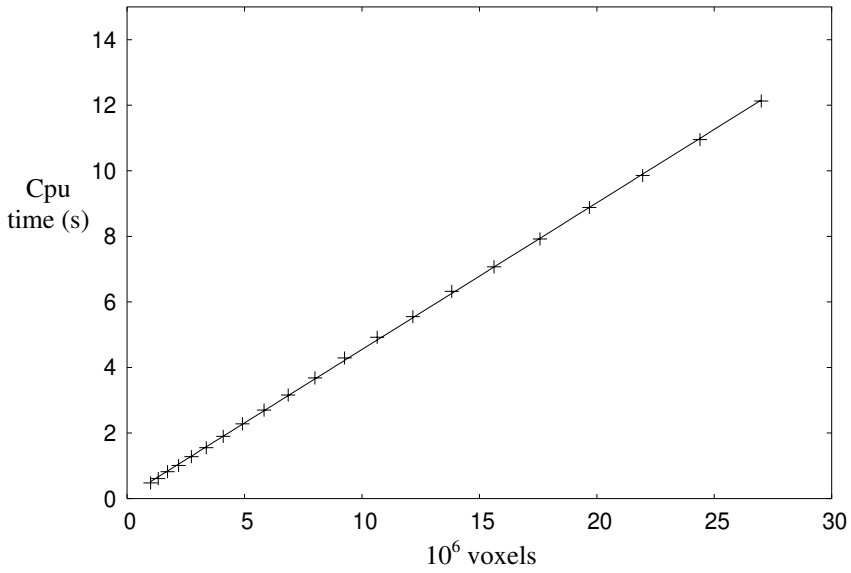


Fig. 1. Cpu time as function of 3D image size in voxels.

4 Reconstruction

Once done with previous steps, each region of the segmented 3D image is delimited with a set of edge points from which we can get the region surface using appropriate reconstruction method. In practice, even for modest regions, the number of points can be quite large (for example a 20 pixel diameter cylinder across 10 slices counts some 600 points) while large regions may require more than 100 000 points. Following [35], we use 3D Delaunay triangulation (**3DT**). The outcome of **3DT** is a set of pair wise linked edge points, in which each linked triplet forms a triangular face and each face quadruplet forms a tetrahedron. The outer surface or the *crust* of the region of interest is a subset of the set of all triangular faces resulting from **3DT.** If the *crust* is convex that the solution is straightforward: it is the Convex Hull [36]. To tackle the problem of the *Concave Hull*, much elaborated algorithms have been developed

In the reconstruction process, we used principally Nina Amanta's POWERCRUST [37]. We have also used Patrice Koehl's TETRAFOR [38], which has very efficient 3D Delaunay Triangulation (Weighted). Once the crust faces obtained, there is one last step which consists on correctly orienting the normal of each face toward the outside of the object. The success of this last step allows for a 3D rendering with no defects during final visualization of the results.

5 Results

First of all and in order to validate our results, we have tested this method on various test 3D images containing various simple objects (cubes, spheres and cylinders). We have also tested random binary images of various sizes and various ratio black/white. We then segmented a 256 slices 256x256 scaled to 8 bits MRI Brain 3D image and extracted and reconstructed the brain surface as the largest region. Fig.2 shows several slices. Fig.3, represents the outer surface reconstructed using a subset of set of edge points for the largest segmented region representing the whole brain.

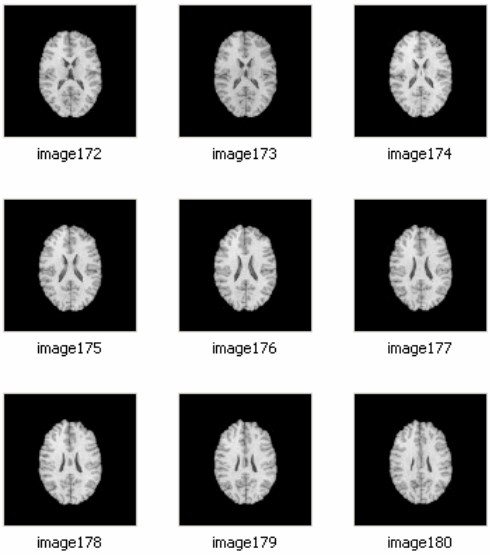


Fig. 2. A set of MRI Brain Image slices.

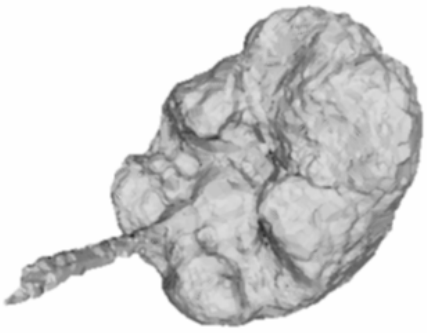


Fig. 3. A bottom view of the whole brain reconstructed from 50% of all the edge points.

Similar results were obtained from 256x256x256 CT Scanner image of the skull of a Monkey. Fig. 4 displays a sample set of slices used.

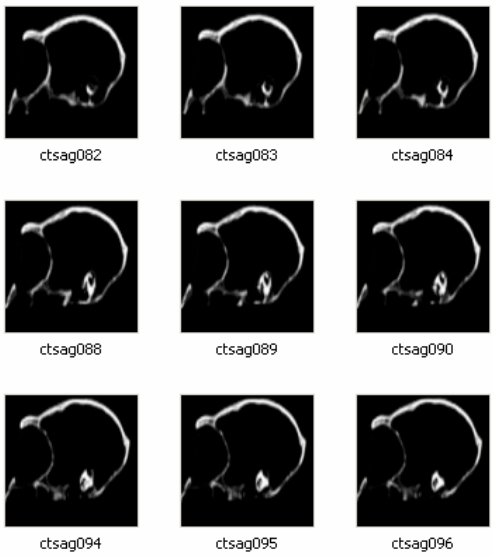


Fig. 4. A set of CT scanner image slices.

The 3D rendering after using half of all edge points as input for powercrust is shown on Fig. 5.

Fig. 5. 3D rendering of crust obtained for the region representing the Skull of a Monkey.

6 Conclusions

We presented a new and very efficient method of 3D segmentation base on both the Enhanced Hoshen-Kopelman algorithm for both lattice and non-lattice environment and applied it to some test images. We illustrate our results by reconstructing the largest region using POWERCRUST. The efficiency of the proposed segmentation approach makes it possible to tackle very large medical images with very small CPU time. An ongoing work is underway to implement this method on parallel machines where better performance is expected.

References

1. O. Monga, I. Bricault. From volume medical images to quadric surface patches. R.R. n°2380. INRIA (1994)
2. C. Revol, M. Jourlin. A new minimum variance region growing algorithm for image segmentation. Pattern Recognition Lett. 18 (1997), 249-258.
3. Mourad Djebaili Segmentation des Images de profondeurs utilisant les analyses multi-resolutions Thèse LIGIM (1998)
4. A. Agnus, C.Ronse, F.Heitz Segmentation spatio temporelle morpho-logique de séquences d'images. LSIIT. UPRES-A 7005, Université de Strasbourg (2000).
5. G. Bueno, O. Musse, F. Heitz, JP. Armspash. Three-dimensional analysis of anatomical structures in MR images on large databases. Université Louis Pasteur (Strasbourg I). LSIIT. CNRS-UPRES- A7005 (2000)
6. M Kass, A. itkin, D. Terzopoulos. Snakes: Active contour models. International Journal of Computer Vision, 1, 312-331 (1988)
7. V.Caselles, F.Catt, C.Coll, and F.Dibos A Geometric model for active contours in image processing Numerische Mathematik, 66: 1-31, 1993
8. R.Malladi, J.A.Sethian, et B.C.Vemuri Shape Modeling with Front Propagation: A Level Set Approach In IEEE Trans on Pattern Analysis and Machine Intelligence volume 17, page 158-175, 1995
9. Jacques-Olivier. Lachaud Extraction de surfaces à partir d'images tridimensionelles : approche discrète par modèle déformable. PhD Thesis, Université de Joseph Fourier (1998)
10. J.O. Lachaud, A. Montavert Deformable meshes with automated topology changes for coarse to fine dimensional surface extraction. Medical Image Analysis (1998) Vol .3, N°2; pp187-207
11. Nicola Scalpel. Optimisation topologique et géométrique d'un maillage surfacique dynamique : application à la segmentation des images médicales volumiques. Rapport de recherche INRIA Sophia-Anitipolis, Projet Epidaure (2000)
12. J. Montagnat, H. Delingette, N Ayache. A review of deformable surfaces: topology, geometry and deformation. Image an Vision Computing 19(2001)1023-1040
13. Isaac Cohen. Modèles déformables 2D et 3D : application à la segmentation d'image médicales. PhD Thesis, Université de Paris IX Dauphine (1992)
14. Johan Montagnat . Modèles déformables pour la segmentation et modélisation d'images médicales 3D et 4D. PhD Thesis Nice Sophia-Antipolis (1999)
15.. Hervé Delingette, Johan Montagnat Extension des maillages simplexes pour la segmentation d'images 3D et 4D Projet Epidaure (1999)
16. Jaques-Olivier. Lachaud Extraction de formes et segmentation d'images volu-métriques par modèle déformable. (1999)

17. X. M. Pardo, M.J. Carreira, A. Mosquera, D. Cabello. A snake for CT image segmentation integrating region and edge deformation. Image and Vision computing 19(2001)461-475
18. J.C.Bezdek, LO.Hall, L.P.Clarke Review of MR image segmentation techniques using pattern recognition, Med Physics, 20, 1033-1047, 1993.
19. C.Jaggi Segmentation par méthode markovienne de l'encéphale humain en imagerie par resonance magnétique : théorie, mise en oeuvre et evaluation. PHD thesis, Université de Caen, 1998.
20. R.O.Duda and P.E.Hart Pattern classification and scene analysis, John Wiley and sons, 1973.
21. J.C.Bezdek, LO.Hall, M.C.Clark, D.B.Goldgof and L.P.Clarke Medical image analysis with fuzzy models, Stat Methods in Medical Research, 6, 191-204, 1997.
22. M. Takatsuka, R.A Jaivis, Encoding 3D structural information using multiple self-organizing feature maps. Image and Vision
23. S. Kobashi, Y.Hata, F. Miyawaki, N. Kamiura Volume- quantization-based neural network approach to 3D MR angiography's image segmentation. Image and vision computing 19 (2001) 185-193
24. Eun Yi Kim, Sang Won Hwang, Se Hyun Park, Hang Joon Kim Spatiotemporal segmentation using genetic algorithms Pattern Recognition 34 (2001) 2063}2066
25. H. Benoit-Cattin, T. Zouagi, C. Odet Une vision fonctionnelle de la segmentation d'images. CREATIS, Unité de Recherches CNRS (UMR5515) (2000)
26. Thierry Geraud. Segmentation automatique des structures cérébrales internes en imagerie par résonance magnétique tridimensionnelle .PhD Thesis, Université de l'ENST Paris (1998)
27. Marc Gelgon Segmentation spatio-temporelle et suivi dans une séquence d'images : application à la structuration et à l'indexation vidéo. PhD thesis Université de RENNES I (1998)
28. Tina Kapur, W. Eric, L . Grimson, William M. Wells and Ron Kikinis. Segmentation of brain tissue from magnetic resonance images. Medical Image Analysis (1996) Vol. 1, N°2, pp 109-127
29. Baillard, P. Hellier, C. Barillot. Segmentation of brain 3D images using level sets and dense registration. Medical Image Analysis 5 (2001) 185-194
30. D. Rey, G.Subsol, H. Delingette, N. Ayache. Automatic detection and segmentation of evolving processes in 3D images: application to multiple sclerosis. Medical Image Analysis 6 (2002)163-179.
31. Rosenfeld, A., Pfalz, J. L., 1966. Sequential operations in digital picture processing. J. ACM 13, 471-494.
32. Hoshen, J., Kopelman, R., 1976. Percolation and cluster distribution. I. Cluster multiple labelling technique and critical concentration algorithm. Physical Review B 14, 3438-3445.
33. J.Hoshen. On the application of the enhanced Hoshen-Kopelman algorithm for Image analysis pattern recognition letters 19 (1998) 575-584.
34. A. Al-Futaisi, T. W. Patzek. Extension of the Hoshen-Kopelman algorithm to non-lattice environments. Physica A 321 (2003) 665-678.
35. K.Kies, N.Benamrane, A.Benyettou. 3D Medical Image Segmentation and Surface Modeling using the power crust. Information Technology Journal 4 (4) : 377-381, 2005.
36. Daniel Tiggemann, Simulation of percolation on parallel computers, PhD Thesis, University of Cologne, May 2001.Baldonado, M., Chang, C.-C.K., Gravano, L., Paepcke,

Un méta-modèle pour l'étude de la cohérence globale des contraintes dans les bases de données

Djamel Berrabah, Faouzi Boufarès, Mohamed Badri

Laboratoire LIPN, Université Paris 13
99, avenue J.B. Clément 93430 Villetaneuse France
{djamel.berrabah, faouzi.boufares, mohamed.badri}@lipn.univ-paris13.fr

Résumé. Le travail de modélisation de données est toujours une activité délicate et nécessite une bonne expérience des concepteurs. Il a pour objectif la construction d'un schéma conceptuel. Ce dernier est un ensemble de structures de données accompagnées d'une liste de contraintes afin de représenter au mieux le réel perçu. Les moyens actuels de définition de ces dernières ne permettent ni de représenter un grand nombre de contraintes, ni d'assurent leur cohérence globale et ainsi de garantir la validité des données. Nous proposons dans cet article une approche permettant d'étudier cette cohérence en détectant et localisant d'éventuels conflits. A cet effet, nous proposons deux approches l'une est basée sur la programmation linéaire et l'autre sur la logique des prédicats. Un méta-modèle est proposé afin de sauvegarder les contraintes et les formaliser selon l'approche utilisée.

Mots clés : Schéma conceptuel, Base de Données, cohérence des contraintes.

1 Introduction

De nos jours, les applications deviennent de plus en plus complexes faisant appel à des données de taille importante issues de sources hétérogènes. Durant l'étape de conception, les spécialistes (concepteurs, analyste, …), malgré leur expertise, risquent de mal *structurer* ces données et/ou de mal définir leurs *contraintes*. UML (Unified Modeling Language) et EER (Extended Entity-Relationship) [2, 3, 8, 10, 13] sont les formalismes les plus utilisés durant l'étape de conception dans un grand nombre de domaines informatiques comme l'intégration des systèmes d'information, la construction d'entrepôts de données hétérogènes ou encore la rétro-conception [15, 22]. Beaucoup d'outils CASE (Computer Aided Software Engineering) utilisent les diagrammes UML et EER pour développer graphiquement le schéma conceptuel (SC) de la base de données (BD) et le convertir en un schéma physique cible dans le langage spécifique d'un SGBD (Système de Gestion de Bases de Données). Un schéma conceptuel est composé de deux types d'ensembles S et C. L'ensemble S des *structures* de données regroupe les classes et les liens les reliant tels les associations/relations et les héritages. L'ensemble C est donné pour définir des règles logiques appelées *contraintes*, celui-ci permet de représenter au mieux le monde réel. Plusieurs types de contraintes ont été présentés dans la littérature comme les contraintes de domaines, ou celles des multiplicités, ou encore les contraintes de participations et d'héritage [6, 7, 9, 18]. Par exemple, les figures 1, 2, 3, et 4 illustrent

d'une manière graphique des exemples simples de schémas conceptuels sous forme de diagrammes de classes UML qui regroupent structures et contraintes.
Alors que chaque catégorie de contraintes est de nos jours bien identifiée, peu de travaux sont faits au sujet de leurs interactions. En effet, des *Conflits* peuvent apparaître entre elles. Dans ce cas, la BD peut être partiellement ou même totalement vide. La *cohérence* globale de ces contraintes doit donc être vérifiée, avant de créer la BD associée à SC, en étudiant le problème de leur satisfaction. Dans la plus part des outils CASE, seules les contraintes de multiplicité sont intégrées. Force est de constater que ces derniers n'offrent ni la possibilité d'exprimer un grand nombre de catégories de contraintes, ni à fortiori les mécanismes nécessaires pour détecter et localiser les conflits éventuels qui peuvent exister. En effet, il est nécessaire de vérifier la validité d'un schéma conceptuel avant sa transformation automatique vers un autre schéma cible. Les contraintes ne peuvent donc plus être laissées de côté. Les erreurs de consistance doivent être détectées au début du cycle de développement des logiciels. Notre approche souligne l'importance de la vérification de la cohérence globale des contraintes. En effet, toute exploration future des données (fouille de données, gestion des connaissances) devrait se baser sur des données valides et donc des contraintes que l'on peut vérifier (satisfiables) et vérifiées.
Dans le but d'aider les concepteurs des BD, notre objectif est donc de proposer une approche pour détecter et localiser les conflits éventuels entre les contraintes. Nous combinons, d'une part, les techniques de programmation linéaire et d'autre part, la logique des prédicats pour valider les schémas conceptuels. Nous proposons, ainsi, un nouvel outil CASE capable de détecter les schémas conceptuels invalides.
Nous proposons la formalisation de certaines contraintes sous forme de systèmes linéaires. La résolution de ces derniers permet de décider de la validité de SC. Une autre approche, basée sur la logique des prédicats, permet de traiter un plus grand nombre de familles de contraintes.
Ce papier est organisé comme suit. Dans la deuxième section nous donnons un aperçu sur les contraintes dans la modélisation des BD. Une discussion sur le problème de satisfaction des contraintes est élaborée dans la section 3. Nous définissons les systèmes linéaires possibles pour certaines contraintes. Leur analyse est développée en utilisant les algorithmes de *Fourier-Motzkin* ou de *Branch and Bound* qui sont des techniques de la programmation linéaire. Pour compléter notre étude et afin de prendre en considération le plus grand nombre possible de catégories de contraintes, dans la section 4, nous proposons une nouvelle approche basée sur la logique des prédicats pour étudier les conflits entre les contraintes considérées. Une discussion sur les travaux existants et nos perspectives est donnée dans la section 5 et en conclusion.

2 Les contraintes dans la modélisation des bases de données

Dans le paradigme BD, la modélisation conceptuelle, selon les formalismes UML ou EER, préconise la construction de classes. Ces dernières sont organisées en respectant leurs propriétés. Deux ou plusieurs classes sont reliées entre elles par le moyen d'association binaire ou n-aire. Plusieurs types de relations existent comme les liens d'héritage, d'agrégation ou de composition. Leur représentation graphique est appelée le diagramme de classes UML ou le diagramme EER que nous noterons SC. Les figures 1 et 2 illustrent des exemples simples de SC. Dans le but de préserver la

sémantique des objets du monde réel, le SC est généralement accompagné par des déclarations de contraintes. Les contraintes de multiplicité et celles de clés sont parmi les classes de contraintes les plus populaires. Dans la littérature, d'autres types de contraintes relatives à deux ou plusieurs associations et classes ont été proposées. Elles sont désignées sous le nom de contraintes de niveau supérieur ou des contraintes d'intégrité fonctionnelle, par exemple les contraintes d'*inclusion*, d'*exclusion*, de *simultanéité* et de *dépendances fonctionnelles*. Pour plus de détails concernant ces concepts dans la modélisation des BD on peut voir [6, 7, 11, 12, 16, 17, 19, 21]. La contrainte clé primaire concerne les objets d'une classe indépendamment de tout autre concept, alors qu'une contrainte de multiplicité traite la classe et l'association correspondante, on parle de la participation des objets d'une classe à une association. La déclaration des contraintes revient à rajouter des *informations* dans les SC. Chaque information permet de représenter une contrainte ou une combinaison d'entre elles tels que des *intervalles*, des *ensembles de valeurs*, des mots clés comme *sous-ensemble* ou *xor* (ou exclusif).

Exemple 1 (Exemple d'application : Gestion du Centre National de la Recherche Scientifique CNRS) : (figure1).
Ce schéma conceptuel décrit la gestion des laboratoires qui appartiennent au CNRS. Chaque laboratoire est composé d'équipes. Chacune de ces dernières appartient à un et un seul laboratoire dont les membres (chercheurs) composent les équipes. Dans les laboratoires, les équipes ont à établir des contacts avec d'autres laboratoires. Chaque équipe doit avoir au moins 3 contacts. Les chercheurs travaillent sur des projets.

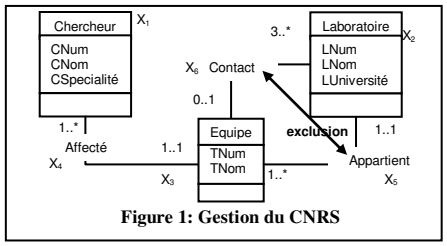


Figure 1: Gestion du CNRS

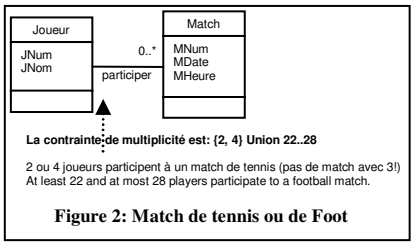


Figure 2: Match de tennis ou de Foot

Exemple 2 : La contrainte de multiplicité peut être un *ensemble de valeurs énumérées* au lieu d'un *intervalle* ou l'union des deux. (figure2)
Dans un match de tennis exactement 2 ou 4 joueurs sont concernés. La contrainte de multiplicité est donc l'ensemble de valeurs {2, 4}, ce qui signifie que le nombre 3 n'est pas permis. Déclarant un intervalle de valeurs [2..4] n'est donc pas correct; et dans un match de football au moins 22 joueurs sont concernés et au plus 28.

Nous avons défini dans [18] le méta-modèle qui nous permet de stocker et de gérer toutes les contraintes. Nous donnons en annexes la version EER (Figure 5).
En Considérant que le schéma conceptuel SC est composé de deux types d'ensembles : S pour les Structures et C pour les Contraints. Les *structures* sont définies par des ensembles qui désignent, d'une part, les classes et leurs propriétés, et d'autre part, les liens entre les classes comme les associations, les agrégations, les compositions et les héritages. Les *contraintes* sont représentées par des tuples comme suit :

- (Cont, Cl, Ass, Label, multiplicité)
- (Cont, Cl, Ass1, Ass2, Label, participation)
- (Cont, Herit1, Herit2, héritage)
- (Cont, Cl1, Cl2, Ass, Label, DF)
- (Cont, Propriété, Opérateur, Valeur)
- (Cont, Propriété, Cl2, Cat_PrimRef, Clé)

où

- Cont, Cl, Ass, Label, Propriété et Herit sont respectivement les noms des contraintes, des classes, des associations, des labels ou rôles, des propriétés et des héritages.
- DF désigne la contrainte de dépendance fonctionnelle
- Opérateur ∈ {<, ≤, =, ≠, ≥, >, In, Not In, …}.

Exemple 3 : Des contraintes définies dans la figure1, elles sont représentées par :

- (k1, Laboratoire, Contact, _, 3..*)
- (k2, Equipe, Contact, _, 0..1)
- (k3, Equipe, Appartient, 1..1)
- (k4, Equipe, Appartient, Contact, Exclusion)
- (k5, Laboratoire, Appartient, Contact, Exclusion)
- (k6, Spécialité, In, {DB, DW, LP, …})
- (k7, RNum, Clé Primaire, _)

3 La satisfaction des contraintes et la programmation linéaire

Afin de représenter au mieux la sémantique du monde réel à modéliser, un SC est, comme indiqué plus haut, un ensemble de structures de données et de contraintes. De nos jours, les outils existants peinent à définir les contraintes. Ils ne permettent ni de représenter un grand nombre de contraintes ni d'assurer à fortiori leur cohérence globale alors que d'éventuels conflits peuvent exister entre elles et ainsi empêcher la vérification de la validité des données. Dans cet article nous soulignons le besoin d'étudier la cohérence globale de ces contraintes pour détecter et localiser les conflits possibles : le problème de satisfaction des contraintes dans les BD volumineuses. Nous formalisons, ci-après, certaines sous forme d'inéquations mathématiques.

3.1 Satisfaction des contraintes de multiplicité

La participation des objets d'une classe E dans une association R consiste à déterminer une contrainte de multiplicité. Nous avons développé dans [16] la notion de multiplicité classique de base qui peut être présentée comme un intervalle de valeurs ou un ensemble de valeurs énumérées. Nous avons défini le concept d'expression de multiplicités qui est une combinaison de conjonctions et de disjonctions de contraintes de multiplicité. Nous résumons ci-dessous ces concepts en utilisant la notation UML :

- Un intervalle **u..v** ; dénote au moins u et au plus v participations des objets de E dans l'association R, avec v ≥ u, u entier non négatif et v entier positif ; les multiplicités **0..0** et **∞..∞** ne sont pas permises. Les multiplicités les plus connues sont : au plus un **0..1** ; exactement un **1..1** ; aucun ou plusieurs **0..*** ; au moins un **1..*** ; au moins u **u..*** ; Les autres interprétations sont : exactement u **u..u** ; au moins u et au plus v **u..v** ; on note que * est un symbole qui désigne **∝** ;
- Une disjonction d'intervalles : 0..u **Union** v..* ; u < v ; au plus u ou au moins v. Cette expression permet de présenter les valeurs qui doivent être prises en compte et celles qui doivent être refusées (figure 2).
- Union d'une disjonction d'intervalles : $u_1..v_1$ **Union** $u_2..v_2$ … **Union** $u_p..v_p$.

- Un ensemble de valeurs énumérées $\{u_1,u_2, u_3, \ldots, u_p\}$ un ensemble fini d'entiers positifs ; son interprétation désigne uniquement les valeurs mentionnées u_1, u_2…ou u_p.
- Union d'intervalles et d'ensembles : u..v **Union** $\{u_1,u_2, u_3, \ldots, u_p\}$.

Soit I une instance (une base de données) d'un schéma conceptuel SC donné. On dit que I satisfait la contrainte de multiplicité K – notée **u..v** – d'une association R sur une class E avec le rôle L, si pour tout objet e de E, le nombre **r** des objets de R connectés à e vérifie $\mathbf{u \leq r \leq v}$ [4, 5, 12, 17, 18].

Soit x_1 le nombre d'objets de E, I(E)=x_1 ; et x_2 le nombre d'objets de R, I(R)=x_2.
Dans le but de satisfaire la contrainte de multiplicité K, la contrainte linéaire
$u\ x_1 \leq x_2 \leq v\ x_1$ doit être vérifiée (x_1 et x_2 sont des entiers positifs).
Quand u = v la contrainte de multiplicité K est satisfaite si la contrainte linéaire
$x_1 - u\ x_2 = 0$ est vérifiée.
A noter que les valeurs 0 et * (* = ∞), pour les multiplicités minimale (u) et maximales (v) ne représentent pas de contraintes.
Etant donné un schéma conceptuel SC, nous lui associons un système linéaire de contraintes Σ dans lequel les variables et les contraintes sont définies comme suit :
- une variable x_i pour chaque classe E de SC ; et une variable x_j pour chaque association R de SC.
- pour chaque contrainte de multiplicité K, tel que u < v est associée l'inéquation linéaire $u\ x_1 - v\ x_2 \leq 0$ si u > 0 et l'inéquation linéaire $x_2 - v\ x_1 \leq 0$ si v < ∞.
- pour chaque contrainte de multiplicité K tel que u = v est associée l'égalité linéaire $x_2 - u\ x_1 = 0$.

A noter que toutes les contraintes du système Σ sont binaires et peuvent être exprimées en $x_2 - u\ x_1 = 0$ ou $u\ x_2 - v\ x_1 \leq 0$ où $u \geq 0$ et $v \geq 0$.

Exemple 4 : Le système linéaire de contraintes ∑ pour le SC de la gestion des projets au CNRS donné dans la figure1 est :

$$\begin{array}{l|l} x_1 - x_4 = 0 & 3.x_2 - x_6 \leq 0 \\ x_2 - x_4 \leq 0 & x_6 - x_3 \leq 0 \\ x_2 - x_5 \leq 0 & x_i > 0 \text{ et entier } \forall\ i=1,6 \\ x_3 - x_5 \leq 0 & \end{array}$$

Nous supposons que les nombres des objets des classes *Chercheur*, *Equipe* et *Laboratoire* sont respectivement x_1, x_2 et x_3. Les nombres d'éléments des associations *Affecté*, *Appartient* et *Contact* sont respectivement x_4, x_5 et x_6.
Le résultat suivant montre l'équivalence entre la validité d'un schéma conceptuel SC et la consistance du système linéaire de contraintes associé.

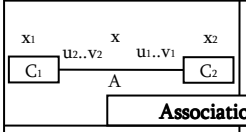
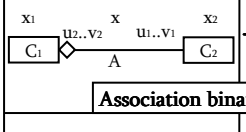
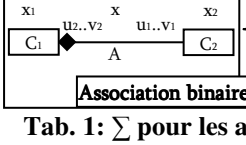
Théorème : Un SC est valide, en respectant ses contraintes, si et seulement si le système linéaire de contraintes ∑ associé a une solution [5, 18].

Preuve : soit E_1, E_2, …, E_n les classes du SC et R_1, R_2, …, R_m ses associations.
SC est valide si et seulement s'il existe une instance (une BD) I de SC qui vérifie toutes ses contraintes. Soit par exemple K – notée u..v – une contrainte de

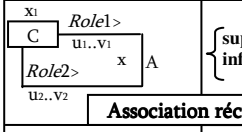
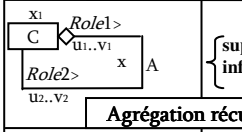
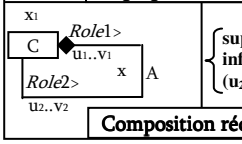
multiplicité de SC entre E_i et R_j et soit $I(E_i)$ et $I(R_j)$ les nombres d'objets respectifs de E_i et R_j dans I. Ayant par définition $u\ I(E_i) \leq I(R_j) \leq v\ I(E_j)$ alors la contrainte linéaire de $\sum$ associée à la contrainte de multiplicité K est vérifiée. Donc, $I(E_1)$, $I(E_2)$, …, $I(E_n)$, $I(R_1)$, $I(R_2)$, …, $I(R_m)$ est une solution de $\sum$.
Soit $X = (x_1, x_2, \ldots, x_n, x_{n+1}, \ldots, x_{n+m})$ une solution de $\sum$. Alors par définition, il existe une instance I de SC tel que $I(E_1) = x_1$, …, $I(E_n) = x_n$, $I(R_1) = x_{n+1}$, …, $I(R_m) = x_{n+m}$ vérifiant toutes les contraintes de multiplicité de SC. ♦

Définition : Un schéma conceptuel SC est valide si et seulement s'il existe au moins une instance (une base de données) qui vérifie toutes ses contraintes.
Si le SC est non valide alors la base de données correspondante sera partiellement ou totalement vide. ♦

L'analyse de la validité d'un schéma conceptuel SC consiste à vérifier la cohérence globale de toutes les contraintes considérées en même temps. En effet, notre objectif est de détecter et localiser les conflits éventuels parmi elles. L'étude de la cohérence globale des contraintes d'un SC nous permet de découvrir si la base de données correspondante sera partiellement vide, totalement vide ou pas.
Plusieurs méthodes de résolution des programmes linéaires (Simplex, Interior points, Fourier-Motzkin, Branch and Bound…) peuvent être utilisées pour résoudre $\sum$. Les détails concernant la programmation linéaire peuvent être retrouvés dans [23, 24, 25, 26]. A noter que pour résoudre le système $\sum$, la méthode que nous adoptons est connue sous le nom de "*Fourier-Motzkin elemination*" [26]. Cette méthode est plus adéquate pour à la fois détecter et localiser les inconsistances. Nous avons donné dans [5] une variante de l'algorithme Fourier-Motzkin. La non existence de solution à $\sum$ devra être suivie par un "retour arrière" dans le processus de résolution où il est possible d'identifier les contraintes qui ont généré les conflits dans SC.
Les tableaux suivants illustrent le système d'inéquations associé correspondant à chaque concept du formalisme UML étudié dans la suite (Tab1, Tab2, Tab3, Tab4, Tab5, Tab6).

Formalisme UML	Système d'inéquations: $\sum$
	$u_1x_1 \leq x \leq v_1x_1$ $u_2x_2 \leq x \leq v_2x_2$
Association binaire	
	$u_1x_1 \leq x \leq v_1x_1$ $u_2x_2 \leq x \leq v_2x_2$
Association binaire : agrégation	
	$u_1x_1 \leq x \leq v_1x_1$ $u_2x_2 \leq x \leq v_2x_2$ $u_2 = 0$ **ou** 1 $v_2 = 1$
Association binaire : composition	

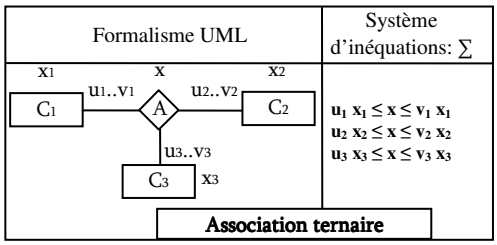
Tab. 1: $\sum$ pour les associations binaires

Formalisme UML	Système d'inéquations: $\sum$
	$\sup(u_1,u_2)\ x_1 \leq x \leq \inf(v_1,v_2)\ x_1$ $\inf(v_1,v_2) \geq \sup(u_1,u_2)$
Association récursive	
	$\sup(u_1,u_2)\ x_1 \leq x \leq \inf(v_1,v_2)\ x_1$ $\inf(v_1,v_2) \geq \sup(u_1,u_2)$
Agrégation récursive	
	$\sup(u_1,u_2)\ x_1 \leq x \leq \inf(v_1,v_2)\ x_1$ $\inf(v_1,v_2) \geq \sup(u_1,u_2)$ $(u_2..v_2 = 0..1$ **ou** $1)$
Composition récursive	

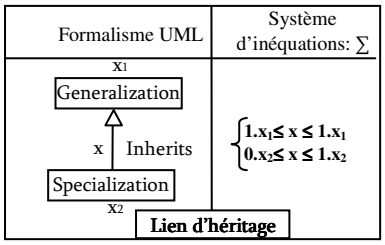
Tab. 2: $\sum$ pour les associations binaires récursives

Notons que :
Si $\mathbf{u_i = v_i}$ alors l'inéquation $\mathbf{u_i\ x_j \le x \le v_i\ x_j}$ devient $\mathbf{u_i\ x_j = x}$.
Si la contrainte de multiplicité est un ensemble valeurs énumérées $\{u_1, u_2, u_3, \ldots, u_p\}$ alors chaque inéquation $\mathbf{u_i\ x_j \le x \le v_i\ x_j}$ devient une des $\mathbf{u_i\ x_j = x}$ avec i=1,p.
Si la contrainte de multiplicité est une disjonction d'intervalles : $\mathbf{u_1..v_1}$ **Union** $\mathbf{u_2..v_2}$ **…** **Union** $\mathbf{u_p..v_p}$ alors p inéquations doivent être vérifiées $\mathbf{u_i\ x_j \le x \le v_i\ x_j}$ avec i=1,p.

Il est facile de voir que les associations récursives avec une multiplicité 1..1 pour l'un des deux rôles est valide si et seulement si l'autre contrainte de multiplicité du deuxième rôle est 1..1. Une classification des relations récursives et des critères qui contribuent à la validité des relations récursives modélisées dans le modèle entité-association a été donnée dans [14].



Tab. 3: ∑ pour l'association ternaire

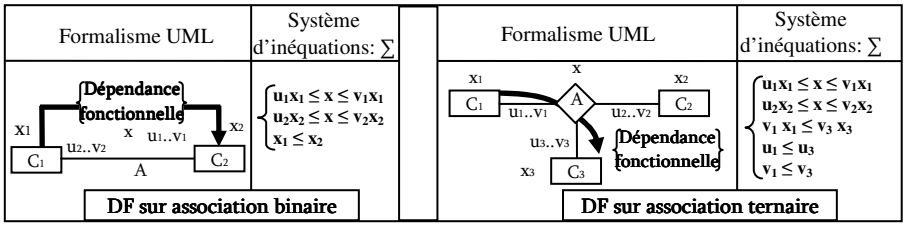


Tab. 4: ∑ pour le lien d'héritage

Nous proposons ici une nouvelle interprétation des associations ternaires en utilisant le système d'inéquations associé [18].
Pour les liens d'héritage, chaque objet appartenant à la classe spécialisée est aussi un élément de celle généralisée [18].

3.2 Satisfaction des contraintes de dépendance fonctionnelle

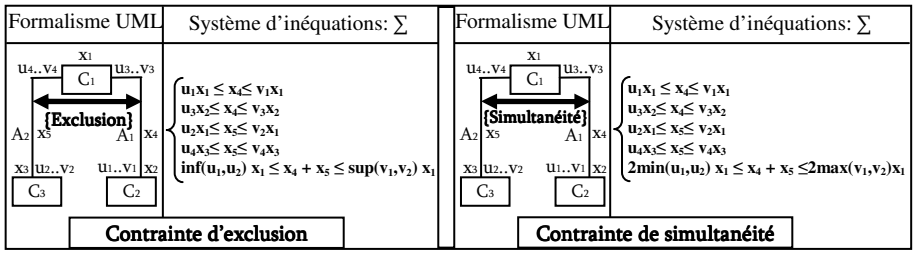
Afin de satisfaire les contraintes de dépendance fonctionnelle (DF), considérons les deux cas (Tab.5). Pour les associations binaires, le nombre des objets de C_2 doit être supérieur à celui des éléments de C_1. Pour les associations ternaires le nombre maximum de la participation de tous les objets de C_1, dans A, est plus petit que $v_3.x_3$ qui est le maximum des occurrences des objets de C_3 dans A.



Tab. 5: ∑ pour la dépendance fonctionnelle

3.3 Satisfaction des contraintes de participations

La satisfaction des contraintes de participation dépend fortement de celle de multiplicité. En effet, le nombre de participation des objets de la classe C_1 dans les associations A_1 et A_2 doit non seulement remplir les conditions du minimum et du maximum de la définition de la multiplicité mais aussi la manière de participation des objets des classes aux associations. Le fait d'appartenir à l'une d'entre elles ou à toutes les deux par exemple. Cette interaction peut générer des conflits. Pour vérifier ces contraintes, les inéquations associées doivent être vérifiées (Tab.6).



Tab. 6: ∑ pour les contraintes de participation

A noter que les systèmes d'inéquations ici ne sont plus binaires. Trois variables sont nécessaires pour traduire les contraintes de participation. Nous proposons une adaptation des algorithmes Branch and Bound afin de résoudre ces systèmes.

Les techniques de la programmation linéaire s'avèrent insuffisantes pour la vérification de la satisfaction des contraintes. En effet, les différentes catégories de contraintes définies dans un SC ne peuvent pas être toutes formalisées moyennant des inéquations. A cet effet, nous proposons une approche alternative basée sur la logique des prédicats.

4. La satisfaction des contraintes et la logique des prédicats

La logique des prédicats est un moyen universel dans la science symbolique et est utilisée par les mathématiciens, les informaticiens et les praticiens de l'intelligence artificielle. Elles est aussi connue sous le nom de la logique du premier ordre [19, 21]. Elle permet de représenter des faits concernant le monde et d'en dériver de nouveaux dans le sens où elle garantit que, si les faits initiaux sont vrais les conclusions sont le sont également. Il s'agit d'un langage formel bien formé, avec une syntaxe et une sémantique bien définies et des règles d'inférences.
Le schéma conceptuel SC, d'une base de données, peut être formalisé en utilisant une base de connaissances (KB) qui est décomposée en une base de faits (FB) et une base de règles (RB). FB est un ensemble de faits représentant toutes les structures et toutes les contraintes du SC. RB est un ensemble de règles représentant les propriétés des structures et des contraintes.

4.1 Base de faits

Un fait est un élément déclaratif qui constitue une information indivisible dans la base de connaissances. Toutes les structures et les contraintes du SC doivent être formalisées comme suit.
Les structures sont définies en utilisant les faits suivants :
propriété (P, Str) : P est la propriété de la structure Str qui peut être une classe ou une association ;
classe (Cl) : Cl représente une classe ;
association (A,Cl) : A est une association dans laquelle Cl participe ;
Les contraintes sont définies en utilisant les faits suivants :
multmin (A,Cl,m_i) : m_i est la valeur de la multiplicité minimum de l'association A associée à la classe Cl ;
multmax (A,Cl,m_a) : m_a est la valeur de la multiplicité maximum de l'association A associée à la classe Cl ;
Les contraintes de participation (exclusion, totalité, simultanéité, inclusion et xor) sont définies sur deux associations A1 et A2 et concernent les classes Cl1 et Cl2. Les faits correspondants sont les suivants.
exclusion (A1,A2,Cl1,Cl2). **totalité** (A1, A2, C1,C2).
simultanéité (A1, A2, C1,C2). **inclusion** (A1, A2, C1,C2).
xor (A1, A2, C1,C2).

Exemple 5 : Quelques faits correspondant au SC donné dans la figure1.
classe (Laboratoire) : ce fait représente une structure qui est la classe *Laboratoire*.
exclusion(assigner,completer,projet,projet) : ce fait représente une contrainte qui exprime qu'un projet p est soit assigné (en cours) soit complété (terminé) mais pas les deux à la fois.

4.2 Base de règles

Une règle est une connaissance qui organise les faits entre eux.
Une règle est notée : *hypothèses* → *conclusions*. Où hypothèses (prémisses) et conclusions (déductions) sont des faits.

Exemple 6 : une règle logique :
SI (∃ **inclusion** entre A1 et A2) et (∃ **exclusion** entre A1 et A2) ALORS (conflit existe).
Hypothèses : exclusion(A1,A2,C1,C2) ^ inclusion(A1,A2,C1,C2)
Conclusions : un conflit existe
En utilisant la notation de Prolog [28, 29] les règles sont notées : ***Conclusions:-hypothèses***.

4.3 Une base de connaissances pour un schéma conceptuel

Nous proposons les règles suivantes pour exprimer les propriétés de toutes les contraintes définies dans un schéma conceptuel SC :
multmin(A,C1,1) ^ multmax(A,C1,1) ^ association(A,C1) ^ association(A,C2) ^ différent(C1,C2) → référence(C1,C2).
reference(C1,C2) → depend(C1,C2).
reference(C1,Cx), depend(Cx,C2) → depend(C1,C2).
reference(C1,Cx), depend(Cx,C1) → cycle(C1).
inclusion(A1,A2,C1,C2) ^ inclusion(A2,A1,C1,C2) → simultanéité(A1,A2,C1,C2).

inclusion(A1,A2,C1,C2) ^ inclusion(A2,A3,C1,C2) → inclusion(A1,A3,C1,C2).
exclusion(A1,A2,_,_) ^ totalité(A1,A3,_,_) ^
inclusion(A3,A2,_,_)→simultanéité(A2,A3,_,_).
exclusion(A1,A2,C1,C2) ^ inclusion(A1,A2,C1,C2) → conflit.
inclusion(A1,A3,_,_) ^ inclusion(A3,A2,_,_) → inclusion(A2,A1,_,_).
exclusion(A1,A2,C1,C2) → exclusion(A2,A1,C1,C2).
totalité(A1,A2,C1,C2) → simultanéité(A2,A1,C1,C2).
xor(A1,A2,C1,C2) → xor(A2,A1,C1,C2).

Exemple 7 : En utilisant la notation Prolog, le SC donné en figure1 est représenté avec la base de connaissances suivante :

classe(chercheur).
classe(equipe).
classe(laboratoire).
association(affecter,chercheur).
association(affecter,equipe).
association(appartenir,laboratoire).
association(appartenir,equipe).
association(contacter,equipe).
association(contacter,laboratoire).
multmin(affecter,chercheur,1).
multmin(affecter,equipe,1).
multmin(appartenir,equipe,1).
multmin(appartenir,laboratoire,3).
multmin(contacter,laboratoire,0).
multmax(affecter,chercheur,10000).
multmax(appartenir,equipe,10000).
multmax(affecter,laboratoire,1).
multmax(appartenir,equipe, 10000).
multmax(contacter,laboratoire,10000).
multmax(contacter,equipe,1).
exclusion(contacter,appartenir,laboratoire,equipe).
xor(R1,R2,X,Y) :- xor(R2,R1,X,Y).
totalité(R1,R2,X,Y) :- totalité(R2,R1,X,Y).
simultanéité(R1,R2,X,Y) :- simultanéité(R2,R1,X,Y).
simultanéité (R1,R2,_,_) :- inclusion(R1,R2,_,_), inclusion(R2,R1,_,_).
simultanéité (R1,R3,_,_) :- inclusion(R1,R2,_,_), inclusion(R2,R3_,_).
référence(C1,C2) :-multmin(A,C1,1), multimax(A,C1,1), association(A,C1),
association(A,C2), différent(C1,C2).
dépend(C1,C2) :- référence(C1,C2).
dépend(C1,C2) :- référence(C1,Cx), dépend(Cx,C2).
cycle(C1) :- référence(C1,Cx), dépend(Cx,C1).

La détection et la localisation des conflits sont faites en raisonnant sur les faits et les règles générés par la traduction du schéma conceptuel étudié. Ce raisonnement est réalisé en posant des faits "**but**" afin de vérifier la base de connaissances donnée sous forme d'un programme logique (PL).
Nous utilisons l'outil "CHIP" (Constraint Handling In Prolog) [28] pour étudier ce programme logique. Cet outil est capable d'intégrer les environnements où les contraintes logiques sont utilisées en interaction.

5. Etat de l'art

Nous avons montré dans [16,18] que notre méta-modèle, implanté avec SQL, permettait la compilation d'un SC et donc les vérifications syntaxique et sémantique. Contrairement à ce qui est utilisé dans [10], où les algorithmes de transformation

présentés ne prennent pas en compte la multiplicité minimale, nous considérons à la fois les deux bornes minimale et maximale, plus encore des expressions de multiplicités qui sont une combinaison de contraintes basiques (les intervalles et ensembles énumérés de valeurs).
Il a été montré dans [4] que la satisfiabilité d'un schéma ER peut être vérifiée dans un temps polynomial en cherchant une solution pour le système d'inéquations associé. Le schéma conceptuel n'est pas "fortement satisfiable" et il est considéré invalide quand le système d'inéquations ne possède aucune solution. Des informations concernant les contraintes de multiplicité non satisfiables peuvent être retrouvées en découvrant les cycles critiques dans le graphe ER qui correspond au schéma ER. Deux algorithmes sont utilisés, alors qu'un seul suffirait [5]. De plus, les associations récursives ne sont pas réellement considérées car les deus rôles différents pris par chaque entité doivent être pris en compte pour vérifier la validité des contraintes de multiplicité. Une classification des associations récursives, dans le modèle ER, et des critères qui contribuent à leur validation ont été donnés dans [14], l'accent a été, justement, mis sur les rôles. Des travaux théoriques présentés dans [12, 17] ont aussi exposé le problème des interactions entre les trois principales catégories de contraintes notamment celles de dépendances fonctionnelles, de multiplicité de type intervalle et de clés, ils ont souligné l'importance de la cohérence globale entre les contraintes.

6. Conclusion

Un outil CASE est en cours de développement. Il est basé sur notre méta-modèle défini dans [16, 18, 20] pour stocker non seulement les structures correspondantes au schéma conceptuel mais aussi toutes les contraintes déclarées. L'analyse de la validité du SC consiste dans un premier temps en la vérification de la syntaxe conformément au méta-modèle et dans un deuxième temps en les aspects sémantiques pour lesquels nous soulignons l'importance de la vérification de la cohérence globale des contraintes.
Deux approches sont proposées pour détecter et localiser les conflits. L'approche mathématique ne permettant de traiter que certains types de contraintes est complétée par l'approche logique. Quelques comparaisons entre ces deux méthodes doivent être faites afin d'étendre ce travail et d'en faire bénéficier la modélisation et la construction d'entrepôts de données, la transformation et l'intégration de schémas et la rétro-conception par exemple. Les catégories de contraintes que nous avons étudiées couvrent une très grande partie de celles utilisées dans la littérature, cependant il serait intéressant d'étudier de nouvelles catégories de contraintes.

Bibliographie

[1] P. P. Chen, "The Entity-Relationship Model -Towards a Unified View of Data", volume 1, ACM Transactions On Database Systems, March 1976, pages 9-36.
[2] J. M. Smith and D. C. P. Smith, "Database abstractions: Aggregation and Generalization", ACM Transactions On Database Systems", volume 2, N°2, June1977, pages105-133.
[3] B. Thalheim, "The Higher-Order Entity-Relationship Model and (DB)2", MFDBS'89 : 2nd Symposium on Mathematical Fundamentals of Database Systems, LNCS 364, Visegrad, Hungary, June 26-301989, pages 382-397.

[4] M. Lenzerini and P. Nobili, "On the satisfiability of dependency constraints in Entity-Relationship schemata", IS' 90, Information Systems, volume 15, N° 4, 1990, pages 453-461.
[5] F. Boufarès, et H. Bennaceur. Consistency Problems in ER-Schemas for Database systems. Journal of information sciences, special issue, 2004.
[6] B. Thalheim, "Fundamentals of Multiplicity Constraints", ER'92 : Entity-Relationship Approach, 11th International Conference on the Entity-Relationship Approach, LNCS 645, Karlsruhe, Germany, October 7-9 1992, pages 7-23.
[7] J. B. Behm and T. J. Teorey, "Relative Constraints in ER Data Models", ER'93 : Entity-Relationship Approach, 12th International Conference on the Entity-Relationship Approach, LNCS 823, Arlington, Texas, USA, December1993 , pages 46-59.
[8] G. Engels, M. Gogolla, U. Hohenstein, K. Hulsmann, P. Lohr-Richter, G. Saake and H. D. Ehrich, "Conceptual modelling of database applications using an extended ER model", Data and Knowledge Engineering, volume 9, 1993, pages 157-204.
[9] A. Rochfeld and P. Negros, "Relationship of relationships and other inter-relationship links in E-R model", Data and Knowledge Engineering, volume 9, 1993, pages 205-221.
[10] R. Elmasri and S. B. Navathe, "Fundamentals of Database Systems", The Benjamin Cummings Publishing Company, Inc., Second Edition 1994 .
[11] O. Rauh and E. Stickel, "Standard Transformations for the Normalization of ER Schemata", IS'96, Information Systems, volume 21, N° 2, 1996, pages 187-208.
[12] S. Hartmann, "On the Consistency of Int-multiplicity Constraints", ER'98: Conceptual Modeling, 17th International Conference on Conceptual Modeling, LNCS 1507, Singapore", November 1998, pages 150-163.
[13] D. Berrabah, F.Boufarès and C.F. Ducateau. "A New Tool to Check the Coherence of Constraints Defined on UML Class Diagrams" IJCSNS International Journal of Computer Science and Network Security, VOL.6 No.8B, August 2006, pp. 162-171.
[14] J. Dullea and Il-Yeol Song, "A Taxonomy of Recursive Relationships and Their Structural Validity in ER Modeling", ER'99, Conceptual Modeling, 18th International Conference on Conceptual Modeling, LNCS 1728", Paris, France, November 1999 , pages 384-398.
[15] J. Akoka, I. Comyn-Wattiau and N. Lammari, "Relational Database Reverse Engineering, Elicitation of Generalization Hierarchies", Advances in Conceptual Modeling, ER'99 Workshops on Evolution and Change in Data Management, LNCS 1727, Paris, France, November 1999 , pages 173-185.
[16] S.K. Lellahi and F. Boufarès, "Meta-Model and loose Semantics for ER-Model", Rapport Technique LIPN N° 2000-3 , mai 2000.
[17] S. Hartmann, "On Interactions of Multiplicity Constraints, Key, and Functional Dependencies", FolKS'2000 : Foundations of Information and Knowledge Systems, First International Symposium, LNCS 1762, Burg, Germany, February 2000, pages 136-155.
[18] D. Berrabah, "Etude de la cohérence globale des contraintes dans les bases de données". Ph. D. Thesis report, CRIP5, Paris 5 University, December 2006.
[19] L. S. Cauman "First-Order Logic: An Introduction". Edition Walter de Gruyter, 1998.
[20] F. Boufarès and N. Kraïem, « A new tool to analyze ER-schemas », The second Asia-Pacific Conference on Quality Software (APAQS 2001), Hong Kong, China, December 10-11, 2001. pp 302-307.
[21] M. Manzano, "Extensions of First Order Logic" Edition Cambridge University Press, 2005.
[22] H. L. Christensen & al. "Operations for Conceptual Schema Manipulation Definitions and Semantics", ADBIS 2001, Advances in Databases and Information Systems, 5th East European Conference, Vilnius, Lithuania, September 2001, LNCS 2151, Proceedings.
[23] C. H. Papadimitriou and K. Steiglitz, "Combinatorial Optimization", Prentice-Hall, Englewood Cliffs, N. J. 1982 ; pp 129-133.
[24] A. Schrijver "Theory of linear and integer programming", Edition Wiley 1986.

[25] G. L. Nemhauser & L. A. Wolsey, "Integer and Combinatorial Optimization" John Wiley and Sons, USA, 1988.
[26] H. P. Williams, "Model Solving in Mathematical Programming" John Wiley and Sons Ltd, England, 1993.
[27] F. Boufarès, H. Bennaceur, A. Abdellatif and F. Sahli « Oracle Designer and the Global Coherence of a Conceptual Data Model », 9th European Conference on IT Evaluation (ECITE'2002), Paris, France, July 15-16, 2002.
[28] M. Dincbas, P. Hentenryck, H. Simonis, A. Aggoun, T. Graf, and F. Berthier, The constraint logic programming lanaguage CHIP, The International Conference on Fifth Generation Computer Systems (FGCS'88), Tokyo, (1988), pp 693-702.
[29] P. Bellot, Objectif Prolog. Ed. Dunod, 1997.
[30] D. Berrabah, F. Boufarès and C.F. Ducateau. "Active Mechanisms for Checking Participation Constraints in UML", 8th International Conference on Enterprise Information Systems (ICEIS-2006), Paphos – Cyprus, 2006, pp. 310-313.

Annexes :

Nous donnons ci-dessous quelques exemples de schémas conceptuels représentant des conflits : conflits générés par interaction de contraintes de participation ; conflits entre héritages et multiplicités ; contradictions dans des définitions de multiplicité …

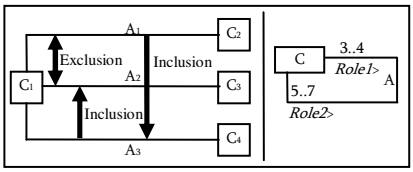


Figure 3 Conflits entre les contraintes de multiplicité et de participation

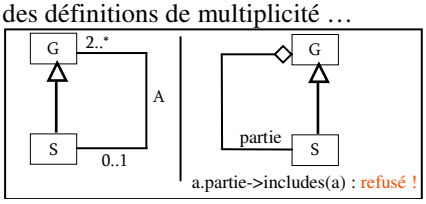


Figure 4 Conflits entre les contraintes de multiplicité et les liens d'héritage

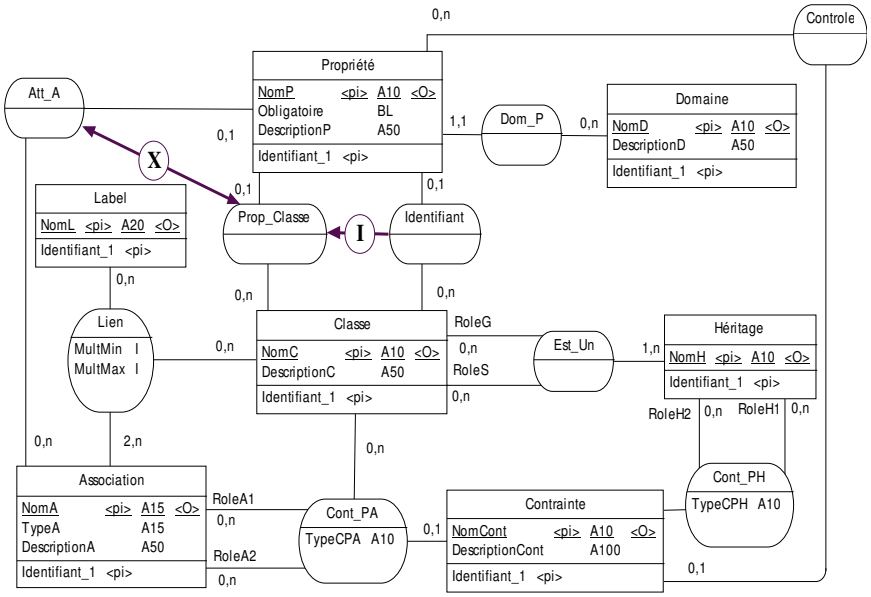


Figure 5 : Méta-modèle pour l'étude de la cohérence globale des contraintes

Algorithme Evolutionnaire Quantique Pour la Recherche d'Ordre de Variables dans les Diagrammes de Décision Binaire

Abdesslem Layeb, Djamel-Eddine Saidouni

Laboratoire lire, université Mentouri de Constantine Algerie
Layeb.univ@gmail.com, saidounid@hotmail.com

Résumé. Dans cet article nous traitons un problème connu dans le domaine des Diagrammes de Décision Binaire [1] qui est le problème d'ordre de variables. Les diagrammes de décision binaire (BDD) servent à représenter de manière symbolique des ensembles d'états. Ils sont largement utilisés dans le domaine de la vérification formelle [2]. L'ordre de variables est une étape très importante dans le processus d'optimisation des BDDs. Un bon ordre de variables va réduire considérablement le nombre d'états dans un BDD ce qui facilite sa manipulation. Malheureusement, l'ordre de variables est un problème d'optimisation combinatoire caractérisé par une grande complexité temporelle et spatiale. Pour cela, plusieurs méthodes basées sur des heuristiques ont été proposées pour trouver le bon ordre de variables d'un BDD. Mais, aucune d'elles n'a résolu le problème efficacement. Pour la résolution de ce problème, on a proposé une nouvelle approche itérative nommée QEABDD basée sur un Algorithme Quantique Evolutionnaire [3]. QEABDD repose sur un noyau de base défini par une représentation quantique appropriée et une dynamique quantique évolutionnaire adaptée. Les résultats obtenus sont encourageants et démontrent la faisabilité et l'efficacité de notre approche pour résoudre ce problème. QEABDD se distingue des autres méthodes évolutionnaires par une taille de population réduite et un nombre raisonnable d'itérations pour trouver le meilleur ordre grâce aux principes de l'informatique quantique tels que la superposition d'états, l'interférence et la mutation quantique.

Mots clés. Optimisation Combinatoire, Diagramme de Décision Binaire, Informatique Quantique, Algorithme Evolutionnaire Quantique.

1. Introduction

L'objectif de la vérification des applications et des circuits électriques est de déceler les erreurs qu'ils contiennent ou de démontrer qu'ils fonctionnent bien. La vérification se fait soit directement sur le système ou bien sur un modèle représentatif de celui-ci. Il existe plusieurs approches de vérification qu'on peut classer en deux classes : les approches informelles (la simulation et les tests) et les approches formelles (la démonstration de théorème et la vérification par évaluation de modèle). La vérification par évaluation de modèle (model-checking en anglais) est une

technique entièrement automatique basée sur les modèles. Elle est largement utilisée pour vérifier des systèmes réactifs.

L'un des problèmes rencontré dans le domaine de la vérification formelle est l'explosion combinatoire. Par exemple dans le modele-checking, le nombre d'états dans les graphes des transitions peut atteindre des niveaux prohibitifs ce qui rend leur manipulation difficile, voire impossible. Pour cela, on utilise des méthodes de compression afin de réduire la taille des graphes d'états. La compression se fait en utilisant des structures de données afin de représenter de façon concise des ensembles d'états. Dans ce cas, Les opérations se font alors sur des ensembles d'états plutôt que sur des états explicites. Cette technique est souvent qualifiée de symbolique.

Parmi les représentations symboliques les plus connues, on trouve la représentation par diagrammes de décision binaire (BDD, de Binary Decision Diagrams). Les BDD sont des structures de données utilisées pour représenter des fonctions booléennes [5]. Les BDD sont largement utilisés dans plusieurs domaines parce qu'ils offrent une représentation canonique et une manipulation facile. Cependant, la taille des BDD dépend de l'ordre de variables choisi. De ce fait, il est important de trouver un ordre de variables qui minimise le nombre de nœuds dans un BDD. Malheureusement, cette tâche n'est pas facile vu qu'on a un nombre exponentiel d'ordres possibles. En effet, Le problème de l'ordre de variables a été démontré NP-difficle [6]. Pour cela, plusieurs méthodes ont été proposées pour trouver le bon ordre de variables d'un BDD et qu'on peut classer on deux classes. La première classe tente d'extraire le bon ordre en inspectant les circuits logiques [7,8]. Par ailleurs, la deuxième classe est basée sur l'optimisation dynamique d'un ordre donné [9,10].

Indépendamment des BDD, les Algorithmes Evolutionnaires Quantiques AEQs [3] constituent un nouveau champ de recherche qui combine les algorithmes évolutionnaires classiques et les principes de l'informatique quantique. Ce dernier est un domaine récent en informatique qui essaie de donner des solutions à des problèmes non encore résolus par l'informatique classique. En effet, l'informatique quantique offre des capacités de traitement et de stockage exponentielles grâce à des principes de la mécanique quantique tels que la superposition, l'enchevêtrement, l'interférence, etc. Cependant, l'utilisation de ce nouveau paradigme est dépendante de la réalisation pratique des machines quantiques. Mais cela n'a pas empêché les chercheurs de combiner les algorithmes classiques et les principes de l'informatique quantique afin de tirer profit des capacités de ce dernier. Contrairement aux algorithmes quantiques purs, les algorithmes inspirés du quantique ne nécessitent pas la présence des machines quantiques. Les algorithmes inspirés du quantique ont démontré leur efficacité dans plusieurs problèmes d'optimisation combinatoire [2, 11,12].

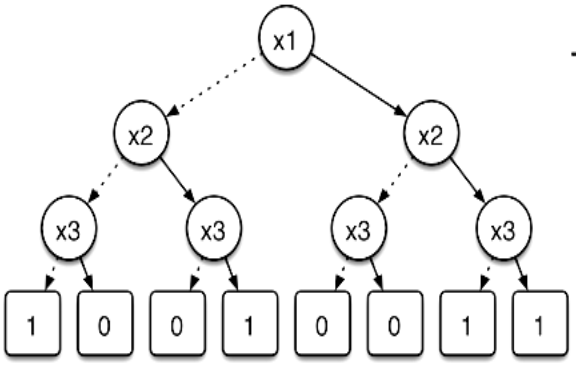
Dans ce contexte, nous proposons dans cet article, une nouvelle approche itérative nommée QEABDD en se basant sur un AEQ. Pour cela, une formulation du problème en terme de représentation quantique a été deriveé et une dynamique évolutionnaire empruntant des opérateurs quantiques a été définie. La représentation quantique des solutions permet le codage de tous les ordres potentiels avec une certaine probabilité. Le processus d'optimisation consiste en l'application d'une dynamique quantique constituée d'un ensemble d'opérations quantiques telles que l'interférence, la mutation quantique et la mesure. Les expériences effectuées sur QEABDD ont montré la faisabilité et l'efficacité de notre approche.

Par conséquent, le reste de l'article est organisé comme suit : La section 2 présente quelques notions de base sur les BDD. Une brève introduction à l'informatique quantique est présentée dans la section 3. L'approche proposée est décrite dans la section 4. La section 5 illustre quelques résultats expérimentaux. Puis, on termine par une conclusion et quelques perspectives.

2. Diagramme de Décision Binaire

Un Diagramme de Décision Binaire ou BDD est une représentation de fonctions booléennes sous forme de graphes orientés acycliques présentant un partage maximal de sous-graphes isomorphes. Il est important de noter que pour un ordre donné de variables, le graphe de décision binaire minimal est unique. La caractéristique de l'unicité permet évidemment de montrer l'équivalence de deux expressions logiques. Pour cette raison, cette structure de données s'est très largement répandue dans plusieurs domaines tel que la microélectronique, la théorie de graphes, l'intelligence artificielle, etc. Récemment, les BDD ont été prouvés efficaces dans le domaine de la vérification formelle des applications "model-checking". En effet, il est théoriquement possible de vérifier toutes les propriétés d'un système sur des représentation à base de BDD [13]. Une fonction booléenne peut être représentée comme un graphe enraciné, dirigé, acyclique, qui se compose des noeuds de décision et de deux nœuds terminaux appelées 0-terminal et 1-terminal (fig.1). La racine et les nœuds intermédiaires sont indexés et possèdent deux noeuds fils appelés *high* et *low*. Un tel BDD s'appelle « ordonné » si les différentes variables apparaissent dans le même ordre sur tous les chemins à partir de la racine. Un BDD peut être réduit en utilisant les deux règles suivantes [13, 14] :

- Eliminer les feuilles et les sous-graphes identiques
- Éliminer les nœuds redondants



x1	x2	x3	*f*
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Fig. 1. Arbre de décision et table de vérité binaires pour la fonction :
f (x1, x2, x3) = - x1 * - x2 * - x3 + x1 * x2 + x2 * x3

Il est très important de prendre en considération l'ordre de variables à utiliser quand on utilise les BDD en pratique (fig.2). Malheureusement, On a un nombre exponentiel d'ordres possibles (permutation). Il est tout a fait claire que le problème d'ordre de variables est un problème NP-difficile [6]. En effet, les algorithmes inspirés du quantique ont démontré leur efficacité dans plusieurs problèmes d'optimisation combinatoire [3, 11,12]. Dans ce contexte, on s'intéresse à appliquer les principes des algorithmes évolutionnaires quantiques sur le problème en cours.

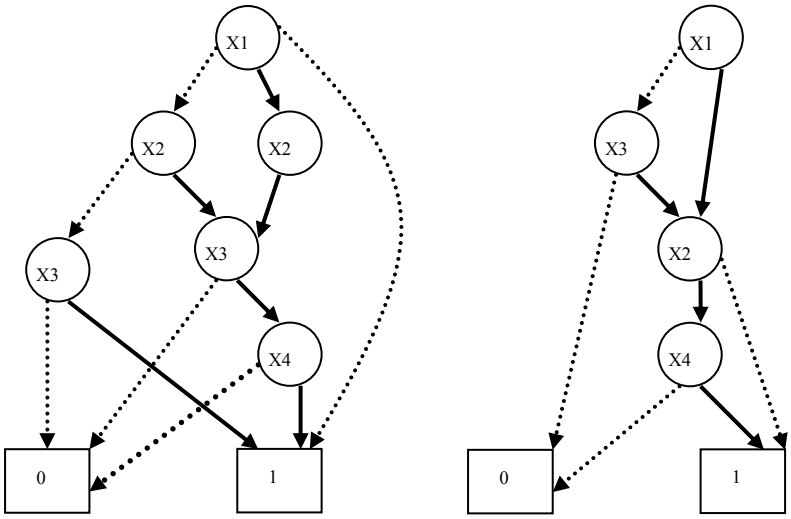


Fig.2. Deux BDD différents représentant la fonction $(x1 \vee x3) \wedge (x2 \Rightarrow x4)$, à gauche l'ordre des variables est : x1, x2, x3, x4 ; et à droite l'ordre est : x1, x3, x2, x4.

3. Informatique Quantique

L'informatique Quantique (IQ) est un domaine émergent faisant appel à plusieurs spécialités : physique, génie, chimie, informatique et mathématiques. L'objectif visé par cette intégration de connaissances est la réalisation d'un ordinateur quantique afin d'effectuer certains calculs beaucoup plus rapidement qu'avec un ordinateur classique. Cette accélération est rendue possible en tirant profit des phénomènes quantiques tels que les superpositions d'états, l'enchevêtrement et l'interférence. Le bit quantique (qubit) est l'unité d'information de base en IQ. Un qubit peut être dans l'état 0, 1, ou une superposition de 1 et 0. Son état peut être donné par [4] :

$$\Psi = a|0\rangle + b|1\rangle \quad (1)$$

où $|0\rangle$ et $|1\rangle$ représentent respectivement les valeurs binaires standard 0 et 1 ; a et b sont les probabilités amplitudes des états correspondants tels que :

$$|a|^2 + |b|^2 = 1 \quad (2)$$

Quand on mesure l'état d'un qubit, on peut avoir l'état 0 avec une probabilité $|a_i|^2$ et on peut avoir l'état 1 avec une probabilité $|b_i|^2$. Cette idée de la superposition permet pour représenter un nombre exponentiel des états en utilisant un nombre restreint de qubits. En attendant la construction de machines quantique, plusieurs chercheurs s'intéressent à la combinaison entre des algorithmes classiques comme les algorithmes évolutionnaires et les principes de la mécanique quantique. L'avantage des ces algorithmes est qu'ils ne nécessitent pas la présence d'une machine quantique.

4. Notre approche

Le développement de l'approche proposée et qu'on désigne par QEABDD repose fondamentalement sur une représentation quantique de l'espace de recherche associé au problème traité ainsi qu'une dynamique utilisée pour explorer cette espace en opérant sur la représentation quantique moyennant des opérations quantiques de base.

4.1. Modélisation de la solution

Le problème d'ordre de variables peut être mathématiquement formulé comme suit :
Ayant un ensemble de variable V={x1, x2,…, Xn}. Le problème est défini par la spécification du couple (Ord, O) où Ord est l'ensemble de tous les ordres possibles et O est une transformation de Ord vers *N* qui donne la taille de BDD correspondant à un ordre donné. Chaque ordre peut être vu comme une permutation de l'ensemble des nombres {1,2…n}. En conséquence, Le problème consiste à trouver la meilleure permutation qui correspond à une taille minimale du BDD.

4.2. Représentation binaire d'un ordre

Afin de pouvoir manipuler les solutions dans un algorithme quantique évolutionnaire, l'ordre de variables est représenté sous forme d'une matrice binaire (fig.3) satisfaisant les contraintes suivantes :

- Pour N variables, la taille de la matrice est N*N, les colonnes représentent les variables et les lignes représentent leurs rangs dans la solution.
- Dans chaque ligne et colonne, on trouve un seul 1 (une variable n'a pas deux rangs et deux variables n'ont pas le même rang).

0	1	0	0
1	0	0	0
0	0	0	1
0	0	1	0

Fig. 3. La représentation binaire matricielle binaire de l'ordre {2, 1, 4,3}.

4.3. Représentation quantique d'un ordre

Comme les algorithmes évolutionnaires classiques, on a utilisé des chromosomes quantiques pour encoder les ordres possibles. Chaque chromosome quantique contient un ensemble de registres quantiques. Chaque registre représente le rang d'un variable dans un ordre, comme il est montré sur la figure 4. Chaque colonne $(a_i \quad b_i)^t$ représente un seul qubit et correspond à un 0 ou 1. Les a_i et b_i sont des valeurs réelles satisfaisant la relation $|a_i|^2 + |b_i|^2 = 1$. Pour chaque qubit une valeur binaire est calculée selon les probabilités $|a_i|^2$ et $|b_i|^2$. Par conséquent, tous les ordres possibles peuvent être représentés par une matrice quantique (fig.5) contenant une superposition de toutes les configurations possibles. Cette matrice quantique peut être vue comme une représentation probabiliste de l'ordre des variables. Cette représentation est efficace quand une approche évolutionnaire est adaptée car elle contribue à la réduction de la taille de la population des chromosomes.

$$\begin{pmatrix} \mathbf{a_1} & \mathbf{a_2} & \cdots & \mathbf{a_m} \\ \mathbf{b_1} & \mathbf{b_2} & & \mathbf{b_m} \end{pmatrix}$$

Fig. 4. Un registre quantique représentant le rang d'une variable

$$\begin{bmatrix} \begin{pmatrix} \mathbf{a}_{11} & \mathbf{a}_{12} & \cdots & \mathbf{a}_{1m} \\ \mathbf{b}_{11} & \mathbf{b}_{12} & & \mathbf{b}_{1m} \end{pmatrix} \\ \begin{pmatrix} \mathbf{a}_{21} & \mathbf{a}_{22} & \cdots & \mathbf{a}_{2m} \\ \mathbf{b}_{21} & \mathbf{b}_{22} & & \mathbf{b}_{2m} \end{pmatrix} \\ \vdots \\ \begin{pmatrix} \mathbf{a}_{n1} & \mathbf{a}_{n2} & \cdots & \mathbf{a}_{nm} \\ \mathbf{b}_{n1} & \mathbf{b}_{n2} & & \mathbf{b}_{nm} \end{pmatrix} \end{bmatrix}$$

Fig. 5. Représentation quantique d'un ordre de variables

4.4. Les opérations quantiques de base

Les opérations quantiques qui sont à la base de la dynamique quantique évolutionnaire proposée sont les suivantes.

– *L'opération de mesure* : Cette opération transforme par projection la matrice quantique en une matrice binaire (fig.6). Donc, on aura une solution parmi toutes les solutions présentes dans la superposition. Mais contrairement à la théorie quantique pure, cette mesure ne détruit pas la superposition. Cela a l'avantage de préserver la superposition pour les itérations suivantes sachant qu'on opère sur des machines classiques. La valeur d'un qubit est calculée selon ses probabilités $|a_i|^2$,

$|b_i|^2$ (équation.1). La matrice binaire est ensuite traduite en une suite de nombres numériques.

- *L'interférence quantique* : Cette opération amplifie l'amplitude de la meilleure solution et diminue les amplitudes des mauvaises solutions. Elle consiste essentiellement à déplacer l'état de chaque qubit dans la direction de la valeur du bit correspondant dans la meilleure solution en cours. L'opération d'interférence permet donc d'intensifier la recherche autour de la meilleure solution. Elle peut être accomplie en utilisant une transformation unitaire qui effectue une rotation dont les angles sont fonction des amplitudes a_i, b_i (Fig.7) et de la valeur du bit correspondant dans la solution référence. Les valeurs de l'angle $\delta\theta$ sont choisies empiriquement de telle manière à éviter une convergence prématurée.
- *La mutation quantique* : Cette opération permet l'exploration de nouvelles solutions et l'augmentation des capacités de diversification du processus de recherche. Elle consiste en l'application d'une perturbation sur un ensemble de qubits de la matrice quantique (Fig.8)

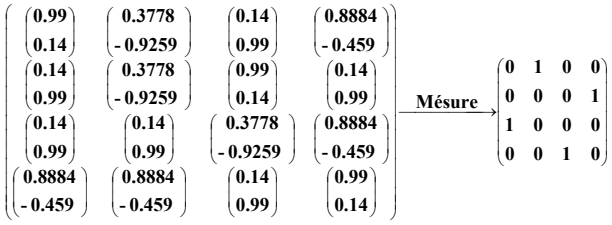


Fig. 6. Projection d'une matrice quantique.

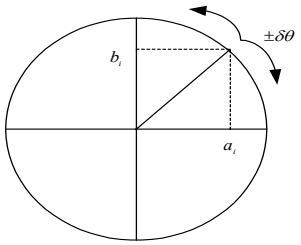


Fig. 7. Interférence quantique

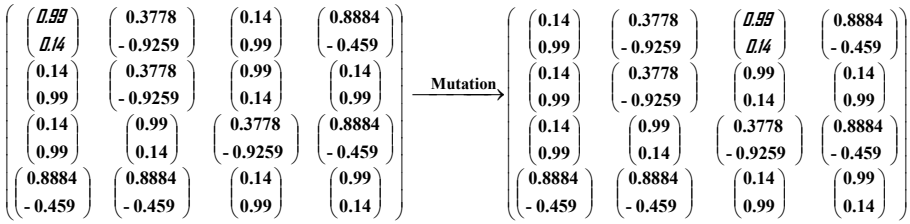


Fig . 8. Opération de Mutation

4.5. Structure et fonctionnement de la méthode QEABDD

QEABDD repose sur un noyau de base défini par une représentation quantique appropriée et une dynamique quantique évolutionnaire adaptée (fig.9). Premièrement, on génère aléatoirement un ordre de variables du BDD. Ensuite, on applique itérativement des raffinements sur cette solution initiale afin d'améliorer sa qualité (fig.9). Pour l'atteinte de notre objectif, on a procédé comme suit. Après l'étape d'initialisation, qui s'agit d'un codage quantique des solutions, le processus consiste en l'application itérative d'opérations d'interférence, mutation, mesure et de sélection (fig.9). La sélection passe par l'évaluation des solutions en utilisant la taille du BDD obtenu comme critère de sélection.

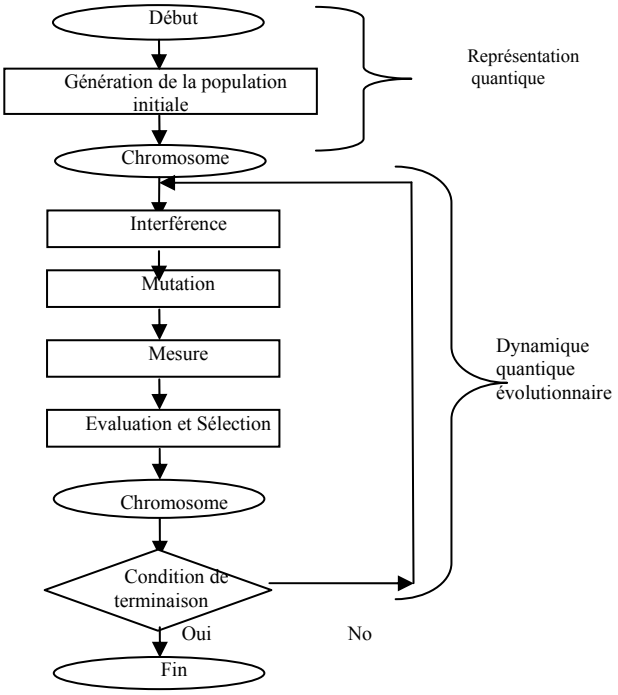


Fig.9. La structure générale de QEABDD

5. Implémentation et évaluation

QEABDD est implémenté en java 1.5 et testé sur un micro-ordinateur avec un processeur de 2 GHZ et 256 MO de mémoire. On a utilisé le package *JAVABDD* [15] qui contient un ensemble d'outils pour la création et la manipulation des BDD. Pour tester la faisabilité de QEABDD on a utilisé plusieurs tests. Les tests sont divisés en deux classes, la première classe contient des fonctions booléennes construites à l'aide

des opérateurs logiques AND, XOR et NOT (table 1). Par ailleurs, la deuxième classe de tests contient des fonctions booléennes construites en utilisant les opérateurs AND, OR et NOT (table 2). Dans chaque table, la deuxième colonne montre le nombre de variables dans le test correspondant, la troisième colonne contient la taille du BDD obtenu à partir de l'ordre initial, et la dernière colonne montre la meilleure taille obtenue du BDD correspondant au meilleur ordre trouvé. Dans toutes les expériences, on a utilisé les paramètres suivants : la taille de la population est *1*, le taux de mutation est *0.15*, l'angle d'interférence est $\pi/20$, et le nombre d'itération est entre *100* pour les petits tests et *1000* pour les grands tests. Les résultats trouvés sont encourageants et démontrent la faisabilité et l'efficience de notre approche.

Table 1. Les résultats obtenus par QEABDD
(les tests contiennent les opérateurs logiques AND, XOR et NOT)

Test	Nombre de variables	Solution initiale	Solution finale
Test1	100	276	151
Test2	100	299	145
Test3	120	330	183
Test4	120	355	190
Test5	130	331	209
Test6	135	381	206
Test7	135	419	211

Table 2. Les résultats obtenus par QEABDD
(les tests contiennent les opérateurs logiques AND, OR et NOT)

Test	Nombre de variables	Solution initiale	Solution finale
Test1	90	132	91
Test2	95	152	98
Test3	100	166	103
Test4	110	170	113
Test5	120	180	124
Test6	130	205	134
Test7	135	225	139

6. Conclusion

Dans ce travail, on a proposé une nouvelle approche nommée QEABDD pour résoudre le problème d'ordre de variable dans les diagrammes de décision binaire. QEABDD est basée sur un algorithme quantique évolutionnaire. La représentation quantique des solutions permet le codage de tous les ordres potentiels avec une certaine probabilité. Le processus d'optimisation consiste en l'application d'une dynamique quantique constituée d'un ensemble d'opérations quantiques telle que l'interférence, la mutation quantique et la mesure. La taille de la population est considérablement réduite grâce au principe de la superposition. Les études

expérimentales prouvent la faisabilité et l'efficacité de notre approche. En effet, QEABDD constitue un noyau de base pour la construction d'un package pour la création et l'optimisation des BDD.

Référence

[1]. S.B. Akers. Binary decision diagrams. IEEE Trans. on Comp., 27:509–516, 1978.

[2]. E. M. Clarke, O. Grumberg, et D. E. Long. Model checking and abstraction. ACM Transactions on Programming Languages and Systems, 16(5) :1512–1542, septembre 1994.

[3]. K.H. HAN and J.H. KIM. "Quantum-inspired Evolutionary Algorithms with a New Termination Criterion, Hε Gate, and Two Phase Scheme," IEEE Transactions on Evolutionary Computation, IEEE Press, vol. 8, no. 2, pp. 156-169, April 2004.

[4]. C.P. Williams and S.H. Clearwater. Explorations in quantum computing. Springer Verlag, Berlin, Germany 1998.

[5]. F. Somenzi, "Efficient Manipulation of Decision Diagrams," in International Journal on Software Tools for Technology Transfer(STTT), vol. 3, pp.171-181, 2001.

[6]. B. Bollig and I. Wegener. Improving the variable ordering of OBDDs is NPcomplete. IEEE Trans. on Comp., 45(9):993–1002, 1996.

[7]. H. Fujii, G. Ootomo, and C. Hori. Interleaving based variable ordering methods for ordered binary decision diagrams. In Int'l Conf. on CAD, pages 38–41, 1993.

[8]. M. Fujita, H. Fujisawa, and N. Kawato. Evaluation and improvements of Boolean comparison method based on binary decision diagrams. In Int'l Conf. on CAD, pages 2–5, 1988.

[9]. U. S. Costa, D. Déharbe, and A. M. Moreira. Variable ordering of bdds with parallel genetic algorithms. In Proceedings of PDPTA'2000, 2000.

[10]. N. Ishiura, H. Sawada, and S. Yajima. Minimization of binary decision diagrams based on exchange of variables. In Int'l Conf. on CAD, pages 472–475, 1991.

[11]. A. Layeb, S. Meshoul and M. Batouhe.: Multiple Sequence Alignment by Quantum Genetic Algorithm. In The 7th International Workshop on Parallel and Distributed Scientific and Engineering Computing of the 20th International Parallel and Distributed Processing Symposium 2006, ISBN: 1-4244-0054-6, pp 1 – 8,Greece, April 2006.

[12]. S. Meshoul, A. Layeb, M. Batouche.: a Quantum Evolutionary Algorithm for effective Multiple Sequence Alignment. In the proc of 12th Portuguese conference on artificial intelligence (EPIA 2005 CMB workshop). Lecture notes in artificial intelligence (LNAI),no 3808, pp.190-201,2005.

[13]. R. Drechsler, et B. Becker. Binary Decision Diagrams: Theory and Implementation. Kluwer Academic Publisher, 1998.

[14]. R. E. Bryant, "Symbolic Boolean Manipulation with Ordered Binary Decision Diagrams", ACM Computing Surveys, Vol. 24, No. 3 (September, 1992), pp. 293-318.

[15]. J. Whaley, JAVABDD, a Java Binary Decision Diagram Library, Stanford University http://javabdd.sourceforge.net/

A Cooperative feature selection approach based on island model

M. A. Esseghir[1,2], T. Hsu[2], G. Goncalves[2], and Y. Slimani[1]

[1] Faculté des Sciences de Tunis
Département des Sciences de l'Informatique
Campus Universitaire, 1060 Tunis, Tunisie.
mohamedemir@gawab.com, yahya.slimani@fst.rnu.tn
[2] Université d'Artois, Faculté des Sciences Appliquées
Laboratoire de Génie Informatique et Automatique de l'Artois (LGI2A)
Technoparc Futura, F 62400 Béthune
hsu@iut-lens.univ-artois.fr, gilles.goncalves@univ-artois.fr

Abstract. As large amounts of valuable data appear in on-line repositories distributed across large-scale networks, a key challenge is to provide services to manipulate this data easily, safely, and with high performance. In the present time we are witnessing an explosive growth in the amount of data that is being collected in the business and scientific areas. Given the size and dimensionality of the datasets, high performance algorithms and systems are an integral component of a successful data mining solution. These data mining solutions try to develop automatic procedures that search these enormous data sets to obtain useful information that would otherwise remain undiscovered. Such new knowledge can take the form of patterns, association rules, clusters, and so on. Feature selection studies how to select a subset of appropriate attributes or variables that are used to build models describing data. In this paper, we propose a new feature selection approach based on the island model as an evolutionary distributed one.
Keywords: Datamining, Feature selection, Curse of dimensionality, Parallel genetic algorithms, Combinatorial optimization, Multi-agents System.

1 Introduction

The increasingly large data sets provided from numerous applications have posed unprecedented challenges to knowledge discovery community and especially in machine learning research. Besides, new data types and structures have emerged with large datasets. Massive data provided or/and found in the web, microArrays experiments make mining tools unable to manage them. Several datamining learning tasks are faced with the problem of selecting the most relevant features to conduct a learning or/and a classification tasks. In this case, human expertise is often required to convert raw data into a set of useful features. We are concerned here by finding automatic feature selection methods. Our work is motivated by the followings: (i) avoiding overfitting; (ii)removing redundant, irrelevant, and noisy data; (iii) yielding comprehensible models; (iv) speeding-up the classifier running and improving resulting knowledge quality (*i.e.* improving learning accuracy and generalization).
In this paper work we investigate specific purposes of metaheuristic optimization strategies and agent computational distribution, especially designed for feature selection problem. To do this, we propose an island wrapper based on collaborative genetic algorithms.
The reminder of this paper is organized as follows. Section 2 formalizes the feature selection problem to be tackled and provides associated definitions. Section 3 reviews the feature selection techniques. Section 4 gives an overview of evolutionary optimization and presents briefely standard Genetic Algorithms and Island model. Section 5 introduces our proposed meta-heuristic optimization model based on agents for feature selection. Section 6 reports results involving eleven benchmark datasets. Finally, we present a summary and we sketch avenues of future research in Section 7.

2 Feature selection problem

Let D be a data set with N attributes such that $\parallel N \parallel = n$, and let M ($M \subseteq N$) be a possible subset of N. Let $J(X)$ the function capable of assessing the relevance of the subset X, that involves the goodness evaluation criterion. The problem of feature selection states the selection of a subset Z such that:

$$J(Z) = max_{X \subseteq N} J(X) \quad (1)$$

In other words, the retained features should be compact and representative of the dataset objects or the studied context as well. This can be done by ignoring redundant and/or irrelevant attributes by keeping the minimal information loss.
One important question that should be addressed here, is how to categorize feature into the following groups: relevant, irrelevant, redundant and noisy features? For this purpose, we can report some existent definitions of relevance [1].
Definition 1.
Strong Relevance: An attribute x_i is strongly relevant if its removal yields a deterioration of the performance in the classification.
Definition 2.
Weak Relevance: An attribute x_i is weakly relevant if it's not strongly relevant and there exists a subset of features X such that the performance on $X \bigcup \{x_i\}$ is better than the performance on X.
Therefore features that are neither strongly relevant nor weakly relevant are irrelevant.
Solutions that have been proposed to tackle this problem have investigated a variety of both evaluation metrics and search approaches [2–4].
The search through the feasible solutions is a combinatorial problem, since a given dataset of n features, implies a search within a feature space of 2^n possible subsets. An exhaustive exploration of the feature space seems to be impractical, especially, when the n became large. In fact, several heuristic strategies have been applied to reduce data dimensionality. Amon these heuristics, we can mention the successful application of the Branch and Bound, probabilistic search and Genetic Algorithms [5–8].
In this paper, we consider the feature selection problem as an optimization problem guided by the accuracy of the selected features. Consequently, we consider that the goal of feature selection is to maximize the accuracy of selected features according to a classification context. A number of relevance measures have been proposed to identify the features having the highest predictive power for a given target. These measures and techniques are presented in the following section.

3 Feature selection techniques

Numerous studies and a plenty of efforts have been done to diversify and improve the feature selection techniques accuracies [4, 9, 10]. These studies methods can be categorized into two methods: *wrappers* and *filters*.

3.1 Filter methods

Considered as the earliest approach to feature selection, filter methods filter out irrelevant features before induction occurs by applying independent search criterion to find appropriate features. FOCUS [11] and RELIEF [12] are considered as the pioneer filter approaches. FOCUS, was originally designed for boolean variables, exhaustively explores the whole set of possible feature subsets by discarding one feature at one time in turn. FOCUS looks for the minimal set of features able to predict pure classes. Nevertheless, RELIEF is a randomized search procedure that attempts to find weak relevant features. Each feature is assigned to a weight that reflects its ability to distinguish among the class values. Features are ranked by weight and those that exceed a threshold parameter are selected to form the

final subset. This method is not well sweeten for datasets with redundant or highly correlated features [9]. The main advantage of the filter methods is their reduced computational time which is due to the *simple* independent criterion used as an evaluation function. However, the biases of the induction algorithm are not taken into account, hence the features selected are, in most of the time, independent of the *classifier* which will pick up these features to use them as inputs.

3.2 Wrapper methods

Comparatively to filters methods, in feature selection based on wrapper methods, the exploration of the feature space is conducted by both the classification accuracy obtained with the subset of selected features and the involved search technique. The classifier is used as a part of the evaluation process by awarding the retained subsets according to their target predictive power. The wrapper methods often have better results than filter ones because they are tuned to the specific interaction between an induction algorithm and its training data [1, 13]. Kohavi *et al.* [10] were the firsts to advocate the wrapper as a general framework for feature selection in machine learning. ID3 [14] and C4.5 [15] were, separately, as used wrappers and both forward selection and backward elimination as heuristic search. On the same way, numerous studies have used the above framework by either changing the wrapper or the search technique. We can remark that recent proposed feature selection techniques have focused on both genetic algorithms and neural networks [1, 6, 8, 16].

Nevertheless, feature selection methods based on wrappers are computationally expensive compared to filters, due to the cost of running the classification algorithm. Some heuristic strategies have been investigated to address the problem of finding the best subsets of features [3, 4].

4 Evolutionary algorithms

4.1 Genetic algorithms

Genetic algorithms (**GAs**) are artificial intelligence global search techniques. They attempt to apply evolutionary techniques to the field of the solving problem, and more specifically in function optimization [17]. They have been proven to be valuable in searching large and complex problem spaces [18], especially for *NP-hard* problems. A GA process is based on natural selection, crossover and mutation that are repeatedly applied to a population of potential solutions, simulating the *Darwinian evolution* [17]. Each solution is encoded into a binary string called *chromosome*. Over generations, a number of a new solutions are generated, with crossover and mutation operations, and the best solutions are selected. This process iteratively continues until the obtention of a good solution. The standard genetic evolution process is sketched by Figure 1.

4.2 Island model

Island model can be viewed as generalization of GA algorithm where GAs are replaced by evolution entities launched in parallel [19].This GA model involves islands where each island evolves one population. It can be also, be defined as an example of a distributed population model where sub-populations are isolated during selection, breeding and evaluation. Isolated in the sense that selection, crossover and mutation are applied independently on the subpopulation.

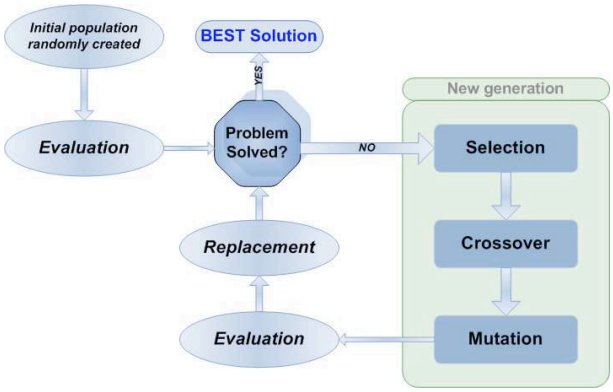


Fig. 1. A common genetic evolution process

5 The proposed approach: dimensionality reduction Island-based model

5.1 Introduction

The main aim of this approach is to improve the global feature selection process quality. This issue is addressed by an evolutionary search strategy. We propose an approach based on island model for the the feature subset selection problem. Our motivation is two folds. Firstly, the island model can be viewed as a boosted genetic algorithm with multi-starting point. In fact, it starts its evolution with independent populations allowing coverage of larger regions in the search space. This feature enables then to evolve. Secondly, the independent evolution of feature selection processes encourage the distribution of computational burden of the whole process, reducing the computational cost. Thirdly, isolated and distributed genetic algorithms can, when evolving independently, share best subsets by solution migration. Consequently, effective and well scaled collaboration leads to search quality improvements.
The proposed model is designed as multi-agent system (MAS), to facilitate independent island evolution and collaboration between them. The choice of an agent-based approach is motivated by both the agents autonomy and collaboration capabilities.

5.2 Multi-Agent Systems

Multi-agent systems (MAS) offer an architecture for distributed problem solving [20]. In fact, agents in MAS need to be proactive and autonomous to perceive their *local* environment, make decision, and behave in a communicative manner with other agents [21]. Particularly, both distributed data mining and problem solving fit, in our sense, very well with the multi-agent framework, since the associated algorithms and solutions require distribution, collaboration and reasoning. Hence, our approach can be viewed as a MAS in the context of evolutionary computation.
In the proposed island model, each GA is embedded into an agent belonging to a container. Hence, these agents mimic the evolutionary process by the behaviors implemented within it. In addition, this MAS provides many collaboration alternatives via agents communication. In the remainder of this paper, container and island will be used interchangeably.

5.3 Genetic algorithm: agent-based version

This section illustrates our first attempts of GA parallelization. In fact, the evolution of the island component is described here. The evolution behavior is now delegated to a set

of agents evolving in a container. Two categories of agents are defined: *main* and *solution* agents. The main agent keeps the whole of GA stages, excepting the fitness evaluation. Since the most computationally cost stage is fitness determination, especially when a classifier is involved in each evaluation, and the evaluation stage of each iteration involves the independent subset assessment, we propose to make evaluation done in parallel. Hence, each time when the main GA needs to evaluate a subset of features, a solution agent is created or/and activated. The main agent communicates the subset to assess to one solution agent. The last one extracts the desired columns, builds the classifiers, and yields the associated fitness to the main agent via a replay message. The evaluations are done by agents in an independent and asynchronous way. Doing so, the evolution processus only waits for the last evaluation.
In both GA and Island model, the chromosomes code the solutions in a binary strings; in these strings each attribute is represented by a bit equal to 1 when it is selected and 0 when it's discarded. Once, the initial random population has been evaluated, the main agent starts the evolution process. The selection procedure which selects reproduction candidates, within current generation, applies *tournament* selection. The common crossover and mutation operators are applied here, whereas the replacement of newly created and accepted chromosomes is done by a *reverse tournament*. The main agent behavior is detailed by Algorithm 1. It starts the evolution with a randomly generated solutions, a classifier, and the set of GA parameters (GA_{params}).

Our island wrapper for feature selection accesses subsets qualities by the prediction level of solutions with two classifiers. The first one is the instance based-learner algorithm ($IB1$). The second is an *ensemble* classifier : the random forest algorithm (RF) [22] which aggregates prediction capabilities of a set of decision trees. It can be considered as a boosted version of C4.5 [15].

Algorithm 1: Agent-GA behavior

Input:
S: Initial solutions set
Cla: Classifier
GA_{params}: Initial GA parameter values
D: Dataset
Output: S' : Population of the last generation

```
1  Begin
2    Population P=S, P_tmp=Ø
3     CreateEvalAgents(P)
4    WaitingLastEval();
5    i=0
6    While i<Maxgen do
7      P_tmp=Select (P, GA_params)
8      Crossover(P_tmp, GA_params)
9      Mutate(P_tmp, GA_params)
10     Evaluate(P_tmp,Cla)
11     WaitingLastEval();
12     Replace(P_tmp, P, GA_params)
13     i=i+1
14   Return (S'=P)
15 End
```

5.4 Island model for feature selection

In the previous section, we presented the evolution within one container, and how the respective agents collaborate to ensure the genetic process. Now, each container will be assigned to an island model. The aggregation of a set of islands with the appropriate communication procedure appoints the model mainstream distribution idea. In fact, starting with more than one population allows diversified explorations and more in depth coverage of the search space, compared to standard GA with one starting point, especially when we deal with exponential large scale growth of search space as in feature selection. Besides, the *independent* island-GA evolution preserves the genetic material particularity and discards fitted solutions dominance risk. Hence, the islands can be viewed like natural species.

Furthermore, both loosely coupled islands and agent design make the whole model well-sweeten for distributed environments. In this case, the computational overflow burden can be shared.

The genetic process within each island will be slightly modified to accept the genetic material exchange with the other islands. Algorithm 2 sketches the collaborative agent-GA behavior. Since the GA, enfolded in each container, has to communicate with other GAs, we define a communication protocol having as purpose to establish the appropriate message exchange rules (i.e. migration frequency, migrant solution(s) number and quality, foreigner integration). All of these parameters are summarized by $Collab_P$ object.

The whole island model process for feature selection is mainly initiated by two steps: the generation of disjoint subsets solutions and the creation of main GA-agents. Both, initialization and end of our island wrapper are detailed in Algorithm 3.

Algorithm 2: Collaborative GA-agent behavior

Input:
S: Initial solutions set
Cla: Classifier
$Collab_P$: Collaboration protocol parameters
GA_{params}: Initial GA parameter values
D: Dataset
Output: S' : Population of the last generation

```
Begin
    Population P=S, P_tmp=Ø
     GenerateEvalAgents(P, D)
    WaitingLastEval();
    i=0
    While i<Maxgen do
        P_tmp=Select (P, GA_params)
        Crossover(P_tmp, GA_params)
        Mutate(P_tmp, GA_params)
        Evaluate(P_tmp, Cla, D)
        WaitingLastEval();
        Replace(P_tmp, P, GA_params)
        M=SelectMigrantSol(P, Collab_P)
        Migrate(M, Collab_P)
        F=GetForeign(Collab_P)
        Integrate(F, P, Collab_P)
        i=i+1
    Return (S'=P)
End
```

Algorithm 3: Island model GA-Agent

Input: N_c : Containers number
Cla: Classifier
$Collab_P$: Collaboration protocol parameters
GA_{params} : Initial GA parameter values
D: Dataset
Output: s : Population of the last generation

```
Begin
    S=GenerateSol(N_c, GA_params )
    i=0
    While i< N_c do
        C=CreateAgentContainer()
        AG=C.CreateGA_Agent(GA_params,S[i], D, Cla, Collab_P )
        C.start()
        i=i+1
    WaitLastAG()
    s=GetBest()
    Return (s)
End
```

6 Empirical study

The aim of this section is to evaluate our approach in terms of classification accuracy and degree of dimensionality reduction. The above described approach is implemented in JAVA using the JADE [23] framework. We show how the island model improves results compared to simple genetic algorithm by varying a set of factors: the number of islands, the classifier and datasets.

6.1 Datasets

For our experiments, we use benchmark datasets provided by the UCI repository [24]. Independently of the features that will be retained in each solution, the chromosomes are evaluated according to the classifier build on the training dataset and accessed with the test set. These two datasets are derived from the original dataset by dividing its instances between these two sets proportionally to the class distribution.

6.2 Dimensionality reduction: GA vs Island-GA

In this section, we compare our island model to GA's in terms of both subsets qualities' and concise representation. In fact, each approach is accessed according to the classification error and to the number of attributes retained by the best chromosome of the population(s) after the evolution process. The classification error metric involves two criteria as defined by the following formula:

$$fitness = (ICI + TMSE)/2 \tag{2}$$

where ICI and $TMSE$ denotes, respectively, the instances misclassification rate of and the mean square error over the test set. Table 6.1 reports the fitness value of the best subsets retained by, respectively, GA and three island configurations. When we compare the island performances toward GA, we can see a slight improvement especially with 5 and 10 containers. Globally, mean values confirm the island models superiority over GAs.
Table 6.2 presents and compares filtered out attributes number for each of the mentioned above experiments (*c.f* Table 6.1).
We can, clearly, depict the effect of feature selection as well on concise representation as on classification accuracy improvement. So, we can keep a concise representation and maintain the same prediction level with only the retained attributes.

Nevertheless, when we compare mean results we can state that island model is well doing in both accuracy and compactness metrics.

	GA		**3 Islands**		**5 Islands**		**10 Islands**	
	IB1	RF	IB1	RF	IB1	RF	IB1	RF
Mushroom	3, 33%	13, 02%	3, 33%	11, 94%	6, 94%	**11,05%**	**2,32%**	11, 30%
Credit	26, 62%	19, 16%	26, 62%	19, 41%	25, 27%	19, 48%	25, 94%	18, 96%
HeartC	23, 46%	20, 15%	23, 46%	19, 38%	22, 31%	19, 38%	22, 31%	19, 38%
Mean	17, 80%	17, 44%	17, 80%	16, 91%	18, 17%	16, 64%	**16,86%**	**16,55%**

Table 1. Performance accuracy

	GA		**3 Islands**		**5 Islands**		**10 Islands**		Dataset
	IB1	RF	IB1	RF	IB1	RF	IB1	RF	
Mushroom	10	9	11	9	9	8	9	8	20
Credit	7	4	8	8	5	3	6	10	15
HeartC	6	10	11	9	6	9	6	9	75
Mean	7, 67	7, 67	10, 00	8, 67	**6,67**	**6,67**	7, 00	9, 00	

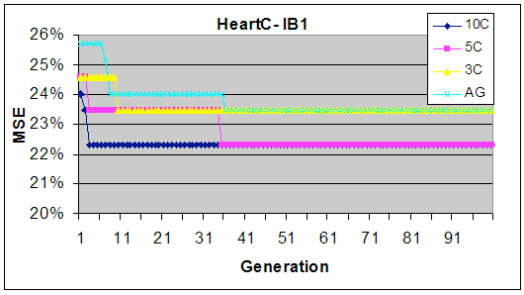
Table 2. Compactness metric performance

6.3 Convergence

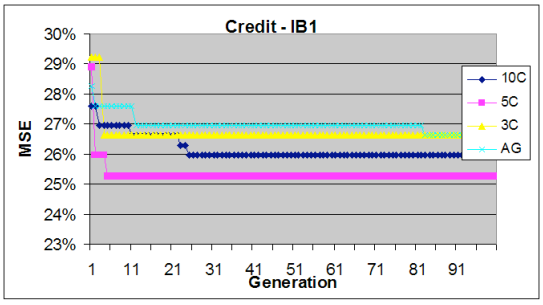
In this section we study the behavior of the evolution process and particularly the fitness evolution over generations and convergence. Figure 2 reports how both GA and three island configurations (3, 5 and 10 containers). Besides, these was applied to the three above described datasets. We can see, in most cases, that our approach, regardless of the number of island involved, was the first to improve the best solution, even when they ends at the same error level. Consequently, island model converges more rapidly than GA's.
Figure 3 shows the results of same experience as the previous one, when we use RF as classifier. Globally, the results confirm the conclusions that we have founded for the IB1 wrapper, though the two involved classifiers don't belong to the same classification family. Finally, we can conclude that the island model outperforms the common GA wrapper for feature selection.

7 Conclusion and perspectives

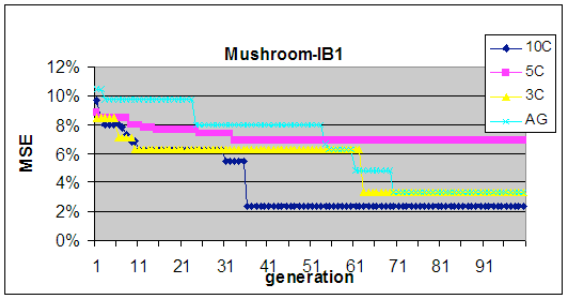
In this paper, we proposed a new feature selection approach based on island model. Then, this proposed method has been evaluated on some benchmark datasets with two classifier families. From these experimentations, we can claim that the design of a feature selection framework based on collaborative of autonomous agents improves the search quality. Nevertheless, more robust and reliable model can be obtained by equipping it an intelligent collaboration schema between and within island models can enhance the performance of the feature selection process. Besides, this work paves the way to large scale distribution and adaptation of high dimensional feature selection problems.



(a) Heart

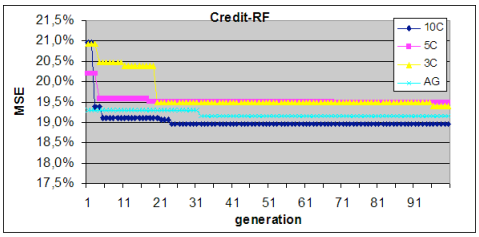


(b) Credit

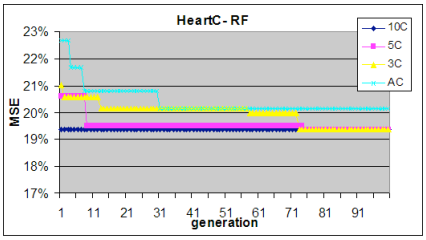


(c) Mushroom

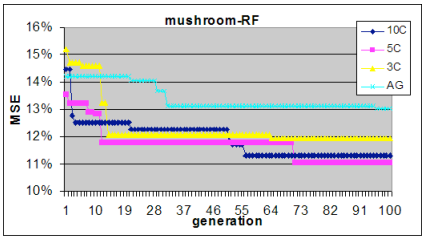
Fig. 2. IB1 classifier as wrapper



(a) Credit



(b) Heart



(c) Mushroom

Fig. 3. RF classifier as wrapper

References

1. Yu, S.: Feature Selection and Classifier Ensembles: A Study on Hyperspectral Remote Sensing Data. PhD thesis, The University of Antwerp (2003)
2. liu, H.: Evolving feature selection. IEEE intelligent Systems (2005) 64–76
3. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. Journal of Machine Learning Research **3** (2003) 1157–1182
4. Guyon, I., Gunn, S., Nikravesh, M., Zadeh, L.: Feature Extraction, Foundations and Applications. Series Studies in Fuzziness and Soft Computing. Springer (2006)
5. Hall, M.A., Holmes, G.: Benchmarking attribute selection techniques for discrete class data mining. IEEE Transactions on Knowledge and Data Engineering **15** (2003)
6. Hong, J., Cho, S.: Efficient huge-scale feature selection with speciated genetic algorithm. Pattern Recognition Letters **27** (2006) 143–150
7. Liu, H., Li, J., Wong, L.: A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns. Genome Informatics **13** (2002) 51–60
8. Huang, C., Wang, C.: A ga-based feature selection and parameters optimization for support vector machines. Expert Systems with Applications **31** (2006) 231–240
9. Kohavi, R., John, G.H.: Wrappers for feature subset selection. Artificial Intelligence **97** (1997) 273–324
10. John, G., Kohavi, R., Pfleger, K.: Irrelevant features and the subset selection problem. In: International Conference on Machine Learning. (1994) 121–129
11. Almuallim, H., Dietterich, T.G.: Learning with many irrelevant features. In: Proceedings of the Ninth National Conference on Artificial Intelligence (AAAI-91). Volume 2., Anaheim, California, AAAI Press (1991) 547–552

12. Kira, K., Rendell, L.: A practical approach to feature selection. In: Proceedings of the Ninth International Conference on Machine Learning (ICML-92) San Francisco, Morgan Kaufmann (1992) 249–256
13. Hall, M.A.: Correlation-based feature selection for machine learning. PhD thesis, Department of Computer Science, University of Waikato, Hamilton, New Zealand (1998)
14. Quinlan, J.R.: Simplifying decision trees. Intl. Journal of Man Machine Studies (1987) 221–234
15. Quinlan, J.R.: Induction of decision trees. In Buchanan, B.G., Wilkins, D.C., eds.: Readings in Knowledge Acquisition and Learning: Automating the Construction and Improvement of Expert Systems. Kaufmann, San Mateo, CA (1993) 349–361
16. Esseghir, M.A., Yahia, S.B., Abdelhak, S.: Localizing compact set of genes involved in cancer diseases using an evolutionary conectionist approach. In: European Conferences on Machine Learning and European Conferences on Principles and Practice of Knowledge Discovery in Databases. ECML/PKDD Discovery Challenge 2005. (2005)
17. Goldberg, D.E.: Genetic algorithms in search, optimization and machine learning. Addison Wesley (1989)
18. Cornujols, A., Miclet, L., Kodratoff, Y., Mitchell, T.: Apprentissage artificiel : concepts et algorithmes. Eyrolles (2002)
19. Engelbrecht, A.P.: Computational Intelligence: An Introduction. InterEditions (2002)
20. da Silva, J.C., Gianella, C., Bhargava, R., Kargupta, H., Klush, M.: Distributed data mining and agents. Engeneering Application of Artificial Intelligence **18** (2005) 791–807
21. Ferber, J.: Les systèmes multi-agents. Vers une intelligence collective. Weiss edn. InterEditions, Paris (1995)
22. Breiman, L.: Random forests. Machine Learning **45** (2001) 5–32
23. JADE: Java Agent DEvelopment framework., (http://jade.cselt.it/)
24. Blake, C., Merz, C.: UCI repository of machine learning databases. (1998). (http :/www.ics.uci.edu/ mlearn/MLRepository.html)

Réconciliation de références : une approche logique adaptée aux grands volumes de données

Fatiha Saïs[1], Nathalie Pernelle[1], and Marie-Christine Rousset[2]

[1] LRI, Université Paris-Sud 11, F-91405 Orsay Cedex,
INRIA Futurs, 2-4 rue Jacques Monod, F-91893 Orsay Cedex, France
{prenom.nom}@lri.fr
[2] LSR-IMAG BP 7238402 St MARTIN D'HERES CEDEX
Marie-Christine.Rousset@imag.fr

Abstract. Le problème de réconciliation de références est un problème majeur pour l'intégration ou la fusion de données provenant de plusieurs sources. Il consiste à décider si deux descriptions provenant de sources distinctes réfèrent ou non à la même entité du monde réel. Cependant, la tâche de réconciliation de références est souvent confrontée à des grandes quantités de données mais aussi à des contraintes de performances en temps d'exécution. En vue d'un passage à l'échelle, nous présentons dans cet article, des techniques que nous intégrons dans notre approche logique L2R de réconciliation de références.

Keywords: Reconciliation de références, efficacité, performance, intégration sémantique de données, ontologies, logique.

1 Introduction

Le problème de réconciliation de références est un problème majeur pour l'intégration ou la fusion de données provenant de plusieurs sources. Il consiste à décider si deux descriptions provenant de sources distinctes réfèrent ou non à la même entité du monde réel (e.g., la même personne, le même article, le même gène, le même hôtel).
Il est très difficile d'attaquer ce problème dans toute sa généralité car les causes d'hétérogénéité dans la description de données provenant de différentes sources sont variées et peuvent être de nature très différente. L'hétérogénéité des schémas est une des cause premières de la disparité de description des données entre sources. De nombreux travaux, dont on peut trouver une synthèse dans [12, 14, 9], ont proposé des solutions pour réconcilier des schémas ou des ontologies par des mappings. Ces mappings peuvent ensuite être utilisés pour traduire des requêtes de l'interface de requêtes d'une source vers l'interface de requête d'une autre source.
L'homogénéité ou la réconciliation de schémas n'empêchent cependant pas les variations entre les descriptions des instances elles-mêmes. Par exemple, deux descriptions de personnes avec les mêmes attributs Nom, Prénom, Adresse peuvent différer sur certaines valeurs de ces attributs tout en référant à la même personne, par exemple, si dans l'un des tuples le prénom est en entier alors que dans l'autre tuple il n'est donné qu'en abrégé.

Dans cet article, nous présentons comment notre méthode logique de réconciliation de références peut s'adapter à de grands volumes de données. Cette méthode est appliquée sur des données décrites relativement à une même ontologie, vue comme un schéma sémantiquement riche, décrit en RDFS étendu par certaines primitives de OWL-DL. OWL-DL sert à poser des axiomes qui enrichissent la sémantique des classes et des propriétés déclarées en RDFS.

Avec l'emergence d'un grand nombre de sources de données sur le web et le besoin de plus en plus accru pour la conception de portails d'informations, les méthodes de réconciliations de références doivent faire face à de très grands volumes de données et à de fortes contraintes de temps d'exécution. Par exemple, les sites comparateurs de prix (e.g. www.kelkoo.com) doivent traiter des millions de références par jour. Certains portails d'information se donnent pour règle de répondre à 30 requêtes en moins de 3 secondes. Ce constat nous conduit à l'obligation de spécifier des méthodes de réconciliation les plus automatiques possible et les moins gourmandes en temps et en espace. Le passage à l'échelle du Web de telles méthodes suppose soit la possibilité de disposer d'un ensemble de connaissances permettant de filtrer les données à réconcilier avant de réaliser des traitements plus complexes pour diminuer l'espace des réconciliations possibles, ou de découper le problème en problèmes qui peuvent être résolus de manière indépendante et éventuellement parallèle.

L'article est organisé de la façon suivante. En section 2 nous définissons le modèle de données, le problème de réconciliation et nous présentons la méthode logique de réconciliation de références. En section 3, nous présentons brièvement des techniques pour améliorer l'efficacité en espace et temps de la méthode de réconciliation. En fin la section 4 conclut l'article.

2 Présentation de l'approche de réconciliation de références

Nous proposons une méthode logique de réconciliation de références qui exploite la sémantique des connaissances du domaine représentées dans le modèle de données RDFS+. Ces connaissance sont traduites en un ensemble de règles qui sont ensuite utilisées par une étape de raisonnement pour inférer des réconciliations et des non réconciliations sûres.

Nous décrivons d'abord le modèle de données que nous avons appelé RDFS+, car il étend RDFS [2] par quelques primitives de OWL-DL [1] et par des règles SWRL[3]. RDFS+ peut être considéré comme un fragment du modèle relationnel (restreint au relation unaires et binaires) enrichi par des contraintes de typage, d'inclusion et d'exclusion entre les relations et d'un ensemble de dépendances fonctionnelles. Ensuite, nous donnerons la représentation des données et enfin nous finirons par une brève présentation de la méthode de réconciliations de références.

2.1 Modèle de données

Le schéma: Un schéma RDFS est composé d'un ensemble de classes (relations unaires) structurées en une taxonomie et d'un ensemble de propriétés (relations binaires) qui peuvent être elles-mêmes structurées en une taxonomie de propriétés. Les propriétés

sont typées. Dans la terminologie RDFS, on distingue les propriétés qui sont des relations, dont les domaines et co-domaines sont des classes, de celles dont le co-domaine est un ensemble de valeurs de base (Integer, date, Literal, ...), et qu'on appelle des attributs.
On notera :

- $R(C, D)$ pour indiquer que le domaine de la relation R est la classe C et que son co-domaine est la classe D, et
- $A(C, Litteral)$ pour indiquer que l'attribut A a comme domaine C et comme co-domaine un ensemble de valeurs (Integer, Date, Literal, ...).

Par exemple, dans le schéma RDFS présenté en figure 1 et correspondant au domaine des lieux culturels, nous avons comme relations *Located(Museum, City), Contains(CulturalPlace, Painting), PaintedBy(Painting, Artist)* et comme attributs *MuseumName(Museum, Literal), YearOfBirth(Artist, Date).*

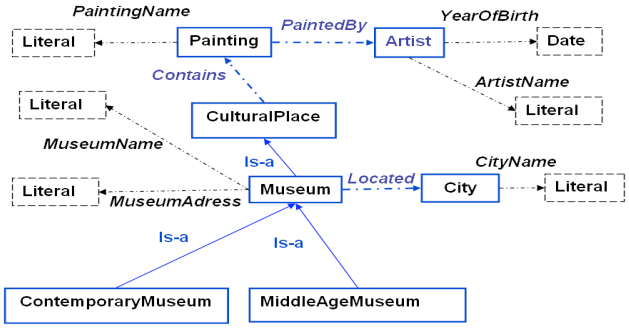


Fig. 1. Exemple de schéma RDFS

Les axiomes du schéma: Nous donnons la possibilité de déclarer des axiomes pour enrichir la sémantique d'un schéma RDFS. Les axiomes que nous considérons sont de plusieurs types.

- **Axiomes de disjonction entre classes.** Nous notons $DISJOINT(C, D)$ l'axiome déclarant que les classes C et D sont disjointes. Par exemple *DISJOINT(Museum,Artist)*.
- **Axiomes de fonctionnalité d'une propriété.** Nous notons $PF(P)$ l'axiome déclarant que la propriété P (relation ou attribut) est fonctionnelle. Par exemple PF(Located) et PF(MuseumName) expriment respectivement qu'un musée est localisé dans une et une seule ville et qu'un musée a un et un seul nom. Ces axiomes peuvent être généralisés à un ensemble $\{P_1, \ldots, P_n\}$ de relations ou d'attributs pour déclarer une combinaison des contraintes de fonctionnalité noté $PF(P_1, \ldots, P_n)$.
- **Axiomes de fonctionnalité de l'inverse d'une propriété.** Nous notons $PFI(P)$ l'axiome déclarant que l'inverse de la propriété P (relation ou attribut) est fonctionnelle. Par exemple $PFI(Contains)$ exprime qu'une peinture ne peut être contenue que dans un seul musée. Ces axiomes peuvent être généralisés à un

ensemble $\{P_1, \ldots, P_n\}$ de relations ou d'attributs pour déclarer une combinaison des contraintes de fonctionnalité inverse noté $PFI(P_1, \ldots, P_n)$. Par exemple, *PFI(PaintingName, PaintedBy)* exprime qu'un artiste et un nom de peinture ne peuvent pas être associés à plusieurs peintures (i.e. les deux sont nécessaires pour identifier la peinture).

– **Axiomes pour les propriétés discriminantes.** Nous notons DISC(A) l'axiome déclarant qu'un attribut A est discriminant. Ce qui signifie que il est connu que toute valeur possible de A a une seule forme (un nombre ou un code). Par exemple les attributs *Year* et *CountryName* peuvent être déclarés comme discriminants.

Il est important de noter que les axiomes de disjonction et de fonctionnalité simple (i.e., de la forme $PF(P)$ ou $PFI(P)$) peuvent être exprimés en OWL-DL alors que les axiomes déclarant les contraintes de fonctionnalité combinées (i.e., de la forme $PF(P_1, \ldots, P_n)$ ou $PFI(P_1, \ldots, P_n)$) et ceux déclarant les propriétés discriminantes (i.e. DISC(P)) nécessitent le pouvoir d'expression du langage SWRL.

Les données: Une donnée a un identifiant (appelé référence) de la forme d'une URI (e.g. *http://www.louvre.fr*, *NS-S1/painting243*), et a une description qui est l'ensemble des faits RDFS qui mentionnent cette référence. Un fait RDFS est :

– soit un fait-classe de la forme $C(i)$ où i est un identifiant,
– soit un fait-relation de la forme $R(i_1, i_2)$ où R est une relation et i_1 et i_2 sont des identifiants,
– soit un fait-attribut de la forme $A(i, v)$ où A est un attribut, i un identifiant et v une valeur (numérique ou alpha-numérique).

Nous considérons que les descriptions de données provenant de différentes sources sont conformes au schéma RDFS+. Afin de distinguer les données provenant de différentes sources, nous utilisons l'identifiant de la source comme préfixe des références. Ces sources de données sont conformes au schema RDFS+ du domaine des lieux culturels (figure 1).

Les axiomes sur les sources de données: Nous considérons deux types d'axiomes. Ils concernent l'hypothèse du nom unique (UNA–Unique Name Assumption) et l'hypothèse du nom unique locale (LUNA–Local Unique Name Assumption). L'UNA déclare que deux données provenant de la même source et ayant deux références différentes réfèrent forcément à deux entités différentes du monde réel (et ainsi elles ne peuvent pas être réconciliées). Une telle hypothèse est valide quand il s'agit d'une source propre (i.e. sans redondances). La LUNA est une hypothèse plus légère que le l'UNA, et déclare que toutes les références associées à une même référence par une relation réfèrent à des entités différentes deux–à–deux du monde réel. Par exemple, à partir des faits $authored(p, a_1)$, ..., $authored(p, a_n)$ provenant de la même source, nous pouvons inférer que les références $a_1, \ldots, a_n$ correspondent à des auteurs distincts de l'article ayant comme référence p.

2.2 Le problème de réconciliation références

Soient S_1 et S_2 deux sources de données ayant le même schéma RDFS+. Soient I_1 et I_2 les deux ensembles d'identifiants de leurs données respectives.

Le problème de réconciliation entre S_1 et S_2 consiste à partitioner l'ensemble $I_1 \times I_2$ des paires de références en 2 sous-ensembles REC et $NREC$ regroupant respectivement les paires de références représentant une même entité, et les paires de références représentant deux entités différentes. Dans la suite de l'article, on utilisera la notation relationnelle plutôt que la notation ensembliste: $Reconcile(i1, i2)$ (respectivement $\neg Reconcile(i1, i2)$) pour $(i1, i2) \in REC$ (respectivement pour $(i1, i2) \in NREC$). Une méthode de réconciliation est totale si elle produit un résultat ($Reconcile(i1, i2)$ ou $\neg Reconcile(i1, i2)$) pour tout couple $(i_1, i_2) \in I_1 \times I_2$.

La *précision* d'une méthode de réconciliation est la proportion, parmi les couples pour lesquels la méthode a produit un résultat (de réconciliation ou de non réconciliation), de ceux pour lesquels le résultat est correct.

Le *rappel* d'une méthode de réconciliation est la proportion, parmi tous les couples possibles de $I_1 \times I_2$, de ceux pour lesquels la méthode a produit un résultat correct.

2.3 Méthode logique de réconciliation de références

La méthode de réconciliation logique que nous décrivons dans cette section est une méthode de réconciliation partielle qui a la caractéristique d'être globale et fondée sur la logique : elle traduit les axiomes du schéma par des règles logiques de dépendances entre réconciliations. L'intérêt d'une approche logique est qu'elle garantit une précision de 100%.

Notre approche consiste à traduire les axiomes associés au schéma, incluant l'UNA (quand elle s'applique) et la LUNA, par des règles logiques, et à appliquer un algorithme de raisonnement pour inférer des réconciliations et des non réconciliations, ainsi que des synonymies et des non synonymies entre valeurs de base qui seront éventuellement conservées dans un dictionnaire.

Génération des règles de réconciliation Traduction de l'hypothèse du nom unique par des règles de non réconciliation:

On introduit les prédicats unaires src1 et src2 pour typer chaque référence en fonction de sa source d'origine (srci(X) signifie que la référence X provient de la source Si). La contrainte de l'UNA au niveau des sources S1 et S2 se traduit par les quatre règles suivantes :

$R1 : src1(X) \wedge src1(Y) \wedge (X \neq Y) \Rightarrow \neg Reconcile(X, Y)$
$R2 : src2(X) \wedge src2(Y) \wedge (X \neq Y) \Rightarrow \neg Reconcile(X, Y)$
$R3 : src1(X) \wedge src1(Z) \wedge src2(Y) \wedge Reconcile(X, Y) \Rightarrow \neg Reconcile(Z, Y)$
$R4 : src1(X) \wedge src2(Y) \wedge src2(Z) \wedge Reconcile(X, Y) \Rightarrow \neg Reconcile(X, Z)$

Les deux premières règles traduisent la non réconciliation de deux références provenant d'une même source. Les deux dernières traduisent le fait qu'une référence provenant d'une source S1 (resp. S2) peut être réconciliée avec au maximum une référence de la source S2 (resp. S1).

Pour toute relation R, l'hypothèse LUNA est automatiquement traduite en deux règles $R11(R)$ et $R12(R)$ suivantes:

$R11(R) : R(Z,X) \wedge R(Z,Y) \wedge (X \neq Y) \Rightarrow \neg Reconcile(X,Y)$
$R12(R) : R(X,Z) \wedge R(Y,Z) \wedge (X \neq Y) \Rightarrow \neg Reconcile(X,Y)$

Traduction des disjonctions entre classes par des règles de non réconciliation: Pour chaque paire de classes C et D déclarées disjointes dans le schéma (DISJOINT(C,D)) ou inférées comme telles par héritage, la règle suivante est générée :

$R5(C,D) : C(X) \wedge D(Y) \Rightarrow \neg Reconcile(X,Y)$

Traduction des axiomes de fonctionnalité par des règles de réconciliation de références et de synonymies de valeurs:

- Pour toute relation R déclarée comme fonctionnelle par un axiome PF(R), la règle R6.1(R) est générée, qui traduit le fait que pour une instance de la classe du domaine de R il existe au plus une instance du co-domaine.

  R6.1(R): $Reconcile(X,Y) \wedge R(X,Z) \wedge R(Y,W) \Rightarrow Reconcile(Z,W)$
- Pour tout attribut A déclaré comme fonctionnel par un axiome PF(A), la règle R6.2(A) est générée, qui exprime que pour une instance de la classe du domaine de A il existe au plus une valeur de base appartenant au co-domaine. Le prédicat binaire $SynVals$ permet d'exprimer que deux valeurs de base sont synonymes. Il est l'équivalent sur les valeurs de base du prédicat Reconcile.

  R6.2(A) : $Reconcile(X,Y) \wedge A(X,Z) \wedge A(Y,W) \Rightarrow SynVals(Z,W)$
- Pour toute relation R déclarée comme fonctionnelle inverse par un axiome PFI(R), la règle règle R7.1(R) est générée, qui exprime que pour une instance de la classe du co-domaine de R il existe au plus une instance du domaine.

  R7.1(R): $Reconcile(X,Y) \wedge R(Z,X) \wedge R(W,Y) \Rightarrow Reconcile(Z,W)$
- Pour tout attribut A déclaré comme fonctionnel inverse par un axiome PFI(A), la règle R7.2(A) est générée, qui traduit le fait que pour deux valeurs de base synonymes appartenant au co-domaine il existe au plus une instance de la classe du domaine.

  R7.2(A): $SynVals(X,Y) \wedge A(Z,X) \wedge A(W,Y) \Rightarrow Reconcile(Z,W)$

  Nous générons automatiquement les règles traduisant les axiomes $PF(P_1,\ldots,P_n)$ de fonctionalité des propriétés et $PFI(P_1,\ldots,P_n)$ de fonctionalité inverse des propriétés composés de plusieurs propriétés. Par exemple, $PF(P_1,\ldots,P_n)$, quand tous les P_i sont des relations, est traduit en la règle :

  $$R7.1(P_1,\ldots,P_n) : \bigwedge_{i\in[1..n]} [P_i(Z,X_i) \wedge P_i(W,Y_i) \wedge Reconcile(X_i,Y_i)] \Rightarrow Reconcile(Z,W)$$

  Si certains P_i sont des attributs, les $Reconcile(X_i,Y_i)$ correspondant doivent être remplacés par $SynVals(X_i,Y_i)$.

  De manière analogue, nous générons des règles traduisant les axiomes $PFI(P_1,\ldots,P_n)$. Quand tous les P_i sont des relations, alors l'axiome est traduit

dans par la règle :

$$R7.2(P_1, \ldots, P_n) : \bigwedge_{i \in [1..n]} [P_i(X_i, Z) \wedge P_i(Y_i, W) \wedge Reconcile(X_i, Y_i)] \Rightarrow Reconcile(Z, W)$$

– Finalement, pour tout attribut A déclaré comme étant discriminant par l'axiome DISC(A), la règle suivante $R8(A)$ est générée :
R8(A): $\neg SynVals(X, Y) \wedge A(Z, X) \wedge A(W, Y) \Rightarrow \neg Reconcile(Z, W)$

Règle de transitivité : cette règle est générée uniquement dans le cas où l'axiome de l'UNA n'est pas déclaré au niveau des sources de données.

R9: $Reconcile(X, Y) \wedge Reconcile(Y, Z) \Rightarrow Reconcile(X, Z)$

Generation de l'ensemble de faits L'ensemble de faits RDF correspondant aux descriptions des données dans les sources S_1 et S_2 est enrichi par un ensemble de faits automatiquement générés :

– de nouveaux faits-classe, faits-relation et faits-attributs obtenus par héritage i.e., en exploitant la relation de subsumption entre les classes et les propriétés, déclarées dans le schéma RDFS: par exemple le fait *ContemporaryMuseum(i)* est présent dans une des sources, le fait-classe $Museum(i)$ et $CulturalPlace(i)$ sont ajoutés à la description de la source;
– des faits de la forme $src1(i)$ et $src2(j)$ pour chaque référence $i \in I_1$ et pour chaque référence $j \in I_2$;
– des faits de synonymie de la forme $SynVals(v_1, v_2)$ pour toute paire (v_1, v_2) de valeurs de base identiques (excepté la ponctuation et les variations de la casse): par exemple, le fait *SynVals("La Joconde", "la joconde")* est ajouté car ces deux valeurs diffèrent uniquement de deux majuscules ;
– des faits de non synonymie de la forme $\neg SynVals(v_1, v_2)$ pour toute paire (v_1, v_2) de valeurs de base distinctes associée a un attribut discriminant. Par exemple, $\neg SynVals(''2004'', '' 2001'')$, $\neg SynVals(''FRANCE'', '' PORTUGAL'')$ sont ajoutés si *Year* et *CountryName* ont été déclarés comme *discriminants*.

Raisonnement. Le raisonnement est appliqué sur l'union $\mathcal{R} \cup \mathcal{F}$ de l'ensemble de règles et de l'ensemble de faits générés automatiquement, comme nous l'avons expliqué précédemment. Le raisonnement doit inférer tous les faits de réconciliation, de non réconciliation, de synonymie et de non synonymie qui sont une conséquence logique de $\mathcal{R} \cup \mathcal{S}$. Le raisonnement est fondé sur la sémantique standard de la logique du premier ordre.

Il est important de remarquer que les règles de reconciliation, bien qu'elles ont des conclusions négatives reste de la forme de clauses de Horn, pour lesquelles ils existe des méthodes de raisonnement qui sont complètes pour l'inférence des impliqués premiers, comme par exemple la resolution SLD [6].

L'algorithme de raisonnement que nous avons implémenté dans notre méthode logique, applique la résolution SLD en trois étapes sur l'ensemble de règles sous la forme de clauses de Horn et sur l'ensemble de faits qui sont des clauses (atomes) closes.

Étape de propositionalisation : nous calculons toutes les résolutions possibles des faits clos dans $\mathcal{F}$ avec des clauses de Horn correspondant au règles dans $\mathcal{R}\backslash\{R9\}$. Cela consist à propager les faits clos dans les règles sauf la règle de transitivité $R9$. Le résultat est un ensemble $\mathcal{P}$ de clauses de Horn complètement instanciées, dans lesquelles les seuls littéraux restant sont de la forme $Reconcile(i,j)$, $\neg Reconcile(i',j')$, $SynVals(u,v)$, ou $\neg SynVals(u',v')$, et chaque atome $Reconcile(i,j)$ ou $SynVals(u,v)$ est considéré comme une variable propositionnelle.

Étape d'inférence propositionnelle : la résolution SLD en propositionel est appliquée sur l'ensemble $\mathcal{P}$ des clauses de Horn *propositionnnelles* obtenues à la première étape.

Étape de transitivité: Cette étape est appliqué seulement si la règle $R9$ est dans $\mathcal{R}$, i.e., sauf si l'axiome de l'UNA n'est pas déclaré au niveau des sources de données. La résolution SLD en logique du premier ordre est appliquée sur les faits clos obtenus à l'étape précédente et sur la forme clausale de la règle de transitivité $R9$.

Il est facile de montrer que tout literal l de la forme $Reconcile(i,j)$, $\neg Reconcile(i',j')$, $SynVals(u,v)$, ou $\neg SynVals(u',v')$: $\mathcal{R} \cup \mathcal{F} \models l$ ssi $\mathcal{P} \models l$

Puisque la résolution SLD appliquée sur des clauses de Horn est complète, et puisque $\mathcal{R} \cup \mathcal{F}$ et $\mathcal{P}$ sont équivalents pour la dérivation des littéraux clos, alors cet algorithme garantit la dérivation de tous les faits de la forme $Reconcile(i,j)$, $\neg Reconcile(i',j')$, $SynVals(u,v)$, ou $\neg SynVals(u',v')$ qui peuvent être logiquement inférés à partir de l'ensemble de règles et de l'ensemble de faits.

D'autres raisonneurs, comme par exemple les raisonneurs de la logique de description, aurait pu être utilisés pour la dérivation des faits de réconciliation. Cependant, les logiques de description ne sont particulièrement pas appropriées pour exprimer certaines règles de reconciliation que nous considérons, qui nécessitent des liaisons explicites de variables. De plus, les raisonneurs des logiques de description existants ne garantissent pas la complétude du calcul d'impliqués premiers.

3 Améliorer l'efficacité de L2R

Dans cette partie nous nous intéresserons à l'efficacité de la tâche de réconciliation de références. Il s'agit d'étudier l'efficacité en espace et en temps de la méthode de réconciliation. Une technique élémentaire pour faire de la réconciliation de référence entre deux ensemble I1 et I2 est d'exécuter une boucle imbriquée de comparaisons de toute référence de I1 avec toutes les références de I2. Malheureusement, une telle stratégie nécessite au total |I1|*|I2| comparaisons. Un autre facteur qui pourrait venir augmenter le coût du calcul de la réconciliation est le coût nécessaire pour une réconciliation.

Nous présentons dans cette section certaines techniques permettant d'améliorer à la fois l'espace et le temps nécessaires pour la réconciliation de références et leurs utilisations possible dans L2R.

3.1 Une meilleure gestion de l'espace

L'espace de réconciliation représente l'ensemble des couples de références qui sont candidats à la réconciliation.

Filtrage. Pour diminuer la taille de l'espace des réconciliations, les méthodes de réconciliations de références peuvent utiliser en pré-traitement des techniques de filtrage. Ces techniques exploitent des connaissances du domaine pour limiter le nombre de couples candidats. Il peut s'agir de :

Méthodes dites de *blocking*. on ne considère que les paires de références qui possèdent une (ou plusieurs) caractéristiques communes, caractéristiques telle que le numéro de ISBN pour les livres, ou encore le nom de famille pour les personnes. Ces techniques ont été introduites par [8] et sont utilisées dans des travaux récents tels que [5]. Si il existe une telle caractéristique qui comporte m valeurs, l'espace des réconciliation est divisé par m.

L'exploitation de connaissances du domaine telles que les disjonctions entre classes. Dans [13] deux références appartenant à deux classes disjointes ne sont pas réconciliables. France telecom nous a fournit un corpus décrivant 562368 hotels. Les disjonctions entre classes d'hôtels de pays différents permettent de réduire l'espace des réconciliations de 67,8%.

L'exploitation de propriétés sur les sources telles que l'Unique Name Assumption. deux références issues d'une source de données qui vérifie l'hypothèse de l'UNA sont forcément distinctes.

Notre méthode L2R [13] de réconciliation a été expérimentée sur le corpus CORA (corpus de citations de publications scientifiques qui a servi de benchmark) qui comporte 6000 références d'articles, de conférences et d'auteurs. L'espace de réconciliation concernant les articles et les conférences contient 2587 références. Nous avons obtenu sur ce corpus un rappel de 64,6% sur l'ensemble REC et de 94,9% sur l'ensemble NREC ce qui nous donne un rappel global de 94,5%. L'espace des réconciliations contient donc 6692569 couples de références à comparer. Utiliser l'année comme propriété discriminante, sachant que les publications existent sur 6 années différentes, permet d'avoir des décisions de non réconciliations pour 21,8% de l'espace de réconciliation. L2R a été également expérimentée sur un jeu de données, fourni par France Télécom R&D, qui comporte sept sources de données contenant des références d'hotels. Dans chacune des sources nous avons l'UNA qui est vérifiée. Nous avons obtenu sur ce corpus un rappel de 54% sur l'ensemble REC et de 75,9% sur l'ensemble NREC ce qui nous donne un rappel global de 75,9%. L'axiom de l'UNA, nous a permis d'inférer des non réconciliations pour 9% de l'espace de réconciliation.

Ces non réconciliations peuvent également être propagées en cours de traitement. Ainsi, si une référence $r1$ d'une source $S1$ a été réconciliée à une référence $r2$ d'une source $S2$ qui possède l'UNA, toutes les autres possibilités de réconciliation de $r1$ avec une référence de $S2$ peuvent être éliminées.

Afin de réduire l'espace de réconciliation, le filtrage peut être effectué à une étape de pré-traitement. Cependant, cela peut empêcher la detection des inconsistances : inférer à la fois $Reconcile(i1, i2)$ et $\neg Reconcile(i1, i2)$, car les paires filtrées ne sont plus considérées par la méthode de réconciliation de références. De plus, nous ne pouvons plus garantir la complétude de notre méthode L2R pour l'inférence des non réconciliations.

Partitionnement des données pour partitioner l'espace de réconciliation. Le partitionnement consiste à diviser *l'espace de réconciliation* en plusieurs sous-ensembles (parties) de taille plus petites.

L'espace de réconciliation est partitionné en plusieurs sous-ensembles de paires de références de taille plus petites de manière à ce que la couverture (le rappel) de la méthode de réconciliation ne soit pas diminuée.

Pour partitionner l'espace des réconciliation, on peut d'abord partitionner les références. Pour assurer la non redondance et la non perte d'information, le partionnement doit satisfaire trois critères (utilisés par [11]):

Complétude: pour toute référence il existe une partition P_i contenant cette référence.

Reconstruction: pour toute source S partitionnée en un ensemble de parties P_i, il existe une opération de reconstruction telle que $S = \Box P_i$ pour tout P_i appartenant à l'ensemble des partitions. Cette opération $\Box$ de reconstruction est à définir en fonction du partitionnement effectué. Par exemple, dans le cas où les partitions sont des composantes connexes d'un graphe, l'opération de reconstruction est la fusion de graphes.

Disjonction: une référence n'est présente que dans une seule partition.

Le graphe G d'un ensemble de références I d'une source de données peut être représenté sous la forme d'un multi-graphe orienté étiqueté dont les sommets V_G sont des références et les arcs E_G sont des relations entre références.

$G = \langle V_G, E_G, R_G \rangle$ où
$V_G = I$, $E_G \subseteq (V_G \times R_G \times V_G)$ et $\quad \langle i1, r, i2 \rangle \in E_G \; ssi \; \exists r, r(i1, i2)$

Considérons que l'ensemble de références est représenté dans un multi-graphe G. Ainsi, l'ensemble des partitions dans l'ensemble de références correspond exactement à l'ensemble de composantes fortement connexes CFC que l'on peut former dans le graphe G. Une composante connexe pour une référence i représente le graphe contenant l'ensemble des références atteignables à partir de i par les relations instanciées.

Une fois que ces CFC ont été définies, elles sont ensuite enrichies par l'ensemble des valeurs atomiques associées aux différentes références par les attributs. Ces derniers sont très importants pour la mesure de la similarité entre les descriptions des références afin de pouvoir décider de leur réconciliation ou de leur non réconciliation. Ainsi, l'espace de réconciliation de référence est réduit à un ensemble d'espaces plus petits correspondant aux paires de composantes fortement connexes du graphe de références G, enrichies par les valeurs des attributs associés à chaque référence des CFCs. Lors de la réconciliation d'une paire de CFCs, l'ensemble des paires de références est le produits cartésien des deux ensembles de références contenues dans les deux CFCs . Bien sûr, certaines paires pourront ne pas être considérées par la suite compte tenu des connaissances du domaine et des informations renseignées.

Nous notons que la taille des composantes connexes est dépendant du niveau de redondance dans les sources (présence de l'UNA). Plus précisément, plus la source est redondante plus les composantes connexes de cette source sont petites. Ainsi, si une source qui décrit un ensemble de références bibliographiques possède l'UNA, chaque

publication est reliée à un ensemble d'auteurs (3 en moyenne) auxquels sont également associées d'autres publications qui font donc parties de la même composante connexe. En revanche, une source telle que CORA ne possède par l'UNA et donc chaque composante connexe se limite à la description d'une publication. CORA qui comporte 6000 références donc peut être découpées en 1295 CFC et donc en 1677025 espaces de 25 références en moyenne.

3.2 Une meilleure gestion du temps

Parallélisation de l'exécution de la réconciliation. Afin d'améliorer les performances en terme de temps d'exécution de la réconciliation, une des solutions est de concevoir un algorithme parallèle. Il s'agit de distribuer la tâche de réconciliation sur un ensemble de processus tout en assurant la non perte d'informations. Pour cela, un ensemble de partitions disjointes peuvent être définies comme présenté précédemment. Dans notre méthode de réconciliation le partitionnement satisfait les contraintes de complétude, de reconstruction et de disjonction, donc il est facile de réaliser le calcul parallèle de la réconciliation de références.

Dans le cas où le partionnement ne peut pas retourner des partitions qui satisfont les trois contraintes, [10] propose d'utiliser deux fonctions qui sont importantes pour la distribution des tâches : la fonction "*scope*" qui permet de distribuer les différentes partitions sur différents processus et la fonction "*responsible*" qui permet d'assurer la non redondance, cela en décidant quel processus est responsable de quelle réconciliation. Une définition plus formelle de ces deux fonctions est donnée dans [10].

Améliorer l'éfficacité d'une réconciliation élémentaire. Traditionnellement, chaque comparison d'une paire de références nécessite la comparaison de l'ensemble des valeurs d'attributs communs et éventuellement la prise en compte des références qui leur sont associées par les relations communes. Avec notre méthode logique de réconciliation nous nous ne occupons pas de cette tâche de calcul de similarité entre les références qui peut être parfois coûteux (méthodes globales). Nous avons uniquement un test d'égalité de valeurs de base des attributs impliqués dans les axiomes du schéma (PF, PFI, DISC). Dans une méthode numérique de réconciliation de références, il est possible de sélectionner un ensemble d'attributs et de relations pour lesquels nous avons une connaissance du domaine exprimant leur pertinence pour le calcul de similarité. Ces connaissances peuvent correspondre aux axiomes de fonctionalité et de fonctionnalité inverse (PF, PFI) des relations et des attributs du schéma RDFS+. Dans [15] on propose un ensemble de techniques pour la réduction de la complexité de la réconciliation d'une paire de références. Une première technique consist à sélectionner un ensemble d'attributs assez pertinents afin de diminuer la dimension en terme d'attributs et de relations des paires de références. Ainsi, ils réduisent le temps de génération du modèle induit qui est un arbre de de décision, utilisé pour classifier les paires de références comme réconciliées ou non réconciliées.

4 Conclusion

Nous avons présenté comment notre méthode logique de réconciliation de références peut s'adapter aux grands volumes de données et à des contraintes strictes de temps.

Nous avons proposé quelques stratégies permettant de limiter la taille des données et le temps de calcul, en vue d'un passage à l'échelle de la réconciliation de références.

Le passage à l'échelle du Web peut s'agir également du fait d'envisager l'application des méthodes de réconciliation dans un contexte pair-à-pair. En d'autres termes, en plus de la distribution et de l'autonomie des sources données, les schémas (ontologies) sont aussi autonome et distribués. Certaines approches travaillent dans ce cadre et estiment que l'utilisation de mappings entre ontologies locales qui se connaissent est plus réaliste que la création d'une ontologie commune importante. Un système tel que *SomeWhere* [4] est une approche complètement pair à pair dans laquelle chaque pair stocke localement ses propres axiomes et un ensemble de mappings. Il exploite les mappings pour répondre de façon correcte et complète à une requête. Une telle approche peut être complétée par une méthode permettant de découvrir et d'exploiter des réconciliations de références afin d'intégrer les données provenant de différents pairs lors d'une requête. [7] proposent une définition des tables de réconciliation (mapping) dans lesquelles sont stockées les références réconciliées entre les différents pairs.

References

1. Owl-dl, http://www.w3.org/2004/owl.
2. Rdfs, http://www.w3.org/tr/rdf-schema/.
3. Swrl, http://www.w3.org/submission/swrl/.
4. Philippe Adjiman, Philippe Chatalic, François Goasdoué, Marie-Christine Rousset, and Laurent Simon. Distributed reasoning in a peer-to-peer setting: Application to the semantic web. *Journal of Artificial Intelligence Research*, 25:269–314, 2006.
5. Rohan Baxter, Peter Christen, and Tim Churches. A comparison of fast blocking methods for record linkage. In *ACM workshop on Data cleaning Record Linkage and Object identification*, 2003.
6. Chin-Liang Chang and Richard Char-Tung Lee. *Symbolic Logic and Mechanical Theorem Proving*. Academic Press, Inc., Orlando, FL, USA, 1997.
7. A. Kementsietsidis, M. Arenas, and R. Miller. Mapping data in peer-topeer systems: Semantics and algorithmic issues, 2003.
8. Howard B. Newcombe and James M. Kennedy. Record linkage: making maximum use of the discriminating power of identifying information. *Commun. ACM*, 5(11):563–566, 1962.
9. Natalya Noy. Semantic integration: a survey on ontology-based approaches. *SIGMOD Record, Special Issue on Semantic Integration*, 2004.
10. Benjelloun Omar, Garcia-Molina Hector, Kawai Hideki, Larson Tait, Menestrina David, and Thavisomboon Sutthipong. D-swoosh: A family of algorithms for generic, distributed entity resolution. In *Technical Report, Stanford InfoLab*, 2006.
11. M. Tamer Ozsu and Patrick Valduriez. *Principles of distributed database systems (2nd ed.)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1999.
12. Erhard Rahm and Philip Bernstein. A survey of approaches to automatic schema matching. *VLDB Journal*, 10:334–350, 2001.
13. Fatiha Sais, Nathalie Pernelle, and Marie-Christine Rousset. Approche logique pour la réconciliation de références. In *Extraction et Gestion des Connaissances conference (EGC 2007)*, 2007.
14. Pavel Shvaiko and Jerome Euzenat. A survey of schema-based matching approaches. *Journal on Data semantics*, 2005.
15. Vassilios S. Verykios, Ahmed K. Elmagarmid, and Elias N. Houstis. Automating the approximate record-matching process. *Information Sciences*, 126(1–4):83–98, 2000.

Learning based system for knowledge discovery from Nasopharyngeal cancer data

Khalid Benabdeslem[1], Mustapha Lebbah[2], Alexandre Aussem[1], Nadjim Chelghoum[1], and Marilys Corbex[3].

[1]University of Lyon1 - LIESP, Lab
8, Boulevard Niels Bohr, 69622 Villeurbanne Cedex, France.
{kbenabde,aaussem,nchelgho}@bat710.univ-lyon1.fr
[2]University of Paris13 - LIMBio, Lab
74, Rue Marcel Cachin, 93017 Bobigny Cedex, France.
lebbah@limbio-paris13.fr
[3]International Agency for Research on Cancer.
150, Cours Albert Thomas, 69280 Lyon Cedex, France.
corbex@iarc.fr

Abstract. In this paper, we present a knowledge discovery system from nasopharyngeal carcinoma (NPC) data. These data have two characteristics: they are qualitative, and present missing values in the database. Considering these two constraints, the system that we propose follows a data mining methodology. It consists in: (1) determining a coding procedure for qualitative data with missing values; (2) Studying the proprieties of topological map of Kohonen and its adaptation for this kind of data; (3) optimizing the number of classes obtained by this algorithm using an automatic clustering method (4) Carrying out decision trees for obtained classes for extracting a subset of rules explaining the NPC and (5) giving semantic interpretations of the extracted profiles associated to classes. Our aim is to find different profiles from the general one. For each profile, we give a subset of explicative variables of NPC from a bi-dimensional map and its associated rules which explain the cancer.

Key words: Qualitative data, SOM, Decision tree, Knowledge discovery

1 Introduction

The Self Organizing Map (SOM) algorithm is an effective tool for the visualization of high-dimensional data. It converts complex, nonlinear statistical relationships between high-dimensional data items into simple geometric relationships on a low-dimensional display. As it thereby compresses information while preserving the most important topological and metric relationships of the primary data items on the display, it may also be thought to produce some kind of abstractions. These two aspects, visualization and abstraction, can be utilized in a number of ways in complex tasks such as process analysis, machine perception, control, and communication [6]. The formalism of the dynamic clustering

provides a general framework which leads to the batch version of the Kohonen's map. In this context, we describe a version of unsupervised neural clustering proposed by [7] dedicated to binary data. This version, called binary topological map (BTM) proposes a clustering of the individual data set $App = \{(z_i, p_i), i = 1..I\}$, where each individual $z_i = (z_i^1, z_i^2, ..., z_i^p)$ is a binary vector of $\beta^d = \{0,1\}^d$ and has a corresponding weight p_i, in our approach the dimension d can be very large (for example about 1000). In this paper, we particularly study the behavior of BTM over qualitative data presenting missing values in the training database. The classes obtained by the clustering paradigm are then presented each one to decision tree algorithm for rules extraction. The data studied here are issued from an epidemiologic study of NasoPharyngeal Carcinoma (NPC). This database is equitably divided on two categories: Cancer and Not-Cancer. In one hand, our aim is two extract profiles of people suffering or not from NPC. In another hand, our study consists on determining, for each extracted profile, its associated set of explicative variables. Subsequently , our work provides a set of features explaining each profile extracted from a global population with the help of an unsupervised clustering done by the binary topological map. This clustering is optimized by an automatic statistical clustering and post-treated by decision trees [4] for explaining the classes by an automatic rules extraction.

The remainder of this paper is organized as follows. Section 2 gives general definitions about qualitative data and their coding. Section 3 presents the binary topological map method. In section 4 we describe the general principle of decision tree algorithm. Finally, the experimentations and the obtained results in Nasopharyngeal carcinoma data analysis are presented in section 5, followed by a conclusion and future works.

2 General information about a binary data

Moreoften binary vector is a coding of discrete variables which have a finite, usually small, number of possible values. Some of these variables, called ordinal variables, have an implicit order, the others are just nominal variables [10] The general coding used in order to obtain binary data are: (a) *The additive binary coding:* This coding respect the order existing between modalities, (b) *The disjunctive complete coding:* which transforms each nominal feature using the disjunctive coding (Table 1.).

`Modalities`	Additive coding	Disjunctive coding
`1`	1 0 0	1 0 0
`2`	1 1 0	0 1 0
`3`	1 1 1	0 0 1

Table 1. Coding of modalities

Euclidian distance is not adapted to binary data, it is often much more interesting to use an appropriate similarity index [8]. In this paper we proceed for the auto organization process with Hamming distance($\mathcal{H}$) proposed in [7] . The distance between two vectors $\mathbf{z}_1$ and $\mathbf{z}_2$ is equal to: $\mathcal{H}(\mathbf{z}_1, \mathbf{z}_2) = \sum_{j=1}^{d} |z_1^j - z_2^j| = b + c$ where b and c are defined in a contingency table (Table 2). It measures the number of mismatch between the two vectors $\mathbf{z}_1$ and z_2.

z_2/z_1	1	0
1	a	b
0	c	d

Table 2. Contingency table, a represents the number of coincident occurance of 1 between the two vectors z_1 and z_2.

A set of individuals in *App* in β^d can be characterized by the central value associated with the Hamming distance, it is the median center. The median center of binary vectors is itself a binary vector whose interpretation is the same as the observations. By definition, the median center of *App* is any point $\mathbf{w} = (w^1, w^2, ..., w^d)$ included in β^d minimizing the inertia of *App*: $\mathcal{I}(w) = \sum_{i=1}^{I} p_i \mathcal{H}(\mathbf{z}_i, \mathbf{w})$, each component w^j minimize $\mathcal{I}(w^j) = \sum_{i=1}^{N} p_i |z_i^j - w^j|$. Thus minimize $\mathcal{I}(w^j) = wP_0 + (1-w)P_1$. Where $P_0 = \sum_{i=1}^{N} p_i(1 - z_i^j)$, represents the sum of weighted observations of which the value is equal to 0, et $P_1 = \sum_{i=1}^{N} p_i z_i^j$ represents the sum of weighted observations of which the value is equal to 1.

To find the median value w^j which minimizes $\mathcal{I}(w^j)$ we choose $w^j \in \{0, 1\}$ so that $\mathcal{I}(w^j)$ has minimum value P_0 or P_1. This returns to take $w^j = 1$ if $P_1 > P_0$ and $w^j = 0$ if $P_1 < P_0$. This rule is simplified when the weights are identical for all variables: w^j is the most often chosen value (0 or 1) by the observations on the variable j.

Missing Values

The problem of the missing values is a real problem in research. There is a different methods to resolve this problem, for example by replacing them by the median value or by the training a prediction system, etc. In our case, the variables are qualitative with fixed number of modalities. We propose then, to define an additional modality for representing the missing values. This solution, does not involve any problem with the disjunctive coding. However, the problem arises using the additive coding, where the order between the modalities is important. For that reason we propose to calculate a median of the variable modalities and to take the whole part of the result like an additional modality.

3 Binary Topological Map

We show now how the use of the median can define a model of self-organizing map adapted to binary data. As for the classical topological maps, we assume that we have a regular grid $\mathcal{C}$, on which we define a distance $\delta(c,r)$, as the length of the unique shortest path on the grid (as undirected graph) between the cells: c and r. The neighborhood system can be defined by a monotonically decreasing kernel function $\mathcal{K}$ ($\mathcal{K} \geq 0$ and $\lim_{|x|\to\infty} \mathcal{K}(x) = 0$). We define the mutual influence of two cells c and r by $\mathcal{K}(\delta(c,r))$. For each cell c of the grid, we associate a referent(weight vector) vector $\mathbf{w}_c = (w^1, ...w^k, ..., w^d)$ of dimension d, where $\mathbf{w} \in \beta^d$. We denote by $\mathcal{W}$ the set of the referents vectors. The topological order is reached using dynamic cluster methods, then we use the cost function defined in [7] dedicated to binary data. This function is equal to:

$$\mathcal{E}(\mathcal{W}, \phi) = \sum_{\mathbf{z}_i \in App} \sum_{r \in C} \mathcal{K}(\delta(\phi(\mathbf{z}_i), r)) \mathcal{H}(\mathbf{z}_i, \mathbf{w}_r) \tag{1}$$

Where ϕ assigns each observation $\mathbf{z}$ to a single cell in $\mathcal{C}$.
The minimization of the cost function (1), is made using an iterative process with two steps:

- **Assignment step**: assuming that $\mathcal{W}$ is fixed, we have to minimize $\mathcal{E}(\phi, \mathcal{W})$ with respect to ϕ. This leads to use the following assignment function:

$$\forall \mathbf{z},\ \phi(\mathbf{z}) = \arg\min_{c} \mathcal{H}(\mathbf{z}, \mathbf{w}_c)$$

- **Optimization step**: assuming that ϕ is fixed, this step minimizes $\mathcal{E}(\phi, \mathcal{W})$ with respect to $\mathcal{W}$ in the space β^d. It is easy to see that this minimization allows to define the median center $\mathbf{w}_c$ of the observations $\mathbf{z}_i \in App$ weihted by $\mathcal{K}(\delta(c, \phi(\mathbf{z}_i)))$. Each component $\mathbf{w}_c = (w_c^1, ..., w_c^k, ..., w_c^d)$ is then computed as follows: $w_c^k = \begin{cases} 0 \text{ if } \left[\sum_{\mathbf{z}_i \in App} \mathcal{K}(\delta(c, \phi(\mathbf{z}_i)))(1 - \mathbf{z}_i^k)\right] \geq \\ \quad \left[\sum_{\mathbf{z}_i \in App} \mathcal{K}(\delta(c, \phi(\mathbf{z}_i)))\mathbf{z}_i^k\right] \\ 1 \text{ otherwise} \end{cases}$,

The minimization of $\mathcal{E}(\phi, \mathcal{W})$ is run by iteratively performing the two steps. At the end $\mathbf{w}_c$, which shares the same binary coding with the observations can be decoded in the same way, allowing a symbolic interpretation of binary referent vectors. The nature of the topological model reached at the end of the algorithm, the quality of the clustering and those of the topological order induced by the graph greatly depend on the neighborhood function $\mathcal{K}$. In practice, binary topological map use smooth function to control the size of the neighborhood as $\mathcal{K}^T(\delta(c,r)) = \exp(\frac{-\delta(c,r)}{T})$. Using this kernel function, T becomes a parameter of the model. As in the classical Kohonen algorithm [6], the preceding is repeated by decreasing T from an initilal value T_{max} to a final value T_{min}.

4 Decision tree

A decision tree is a hierarchical knowledge structure that corresponds to a sequence of decision rules. This method aims to determine which attributes (called explanatory) or which criteria on these attributes provide the best distribution of the actual dataset regardless to a given attribute values (called classes). The tree is built recursively by testing and applying subdivision criteria on a training dataset. The test of criteria is based on a statistical computation of entropy or information gain. There exist diver models of decision trees. Subdivision criteria are determined at attribute level in the ID3 method [11] while they operate on attribute values in CART method [4]. The decision rule sequences are composed of criteria of tree paths starting from the root to the leaves. The main advantage of this technique is its simplicity for decision-makers and people who are not aware of data analysis domain.

5 Results

We apply our developed approach to Nasopharynx Carcinoma data (NPC) issued from an epidemiologic study. To clarify the role of the environment in the etiology of the NPC, the Unit of genetic epidemiology of IARC (International Agency for Research on Cancer) carried out a multicentric study of this kind of cancer in the endemic area of the Maghreb. The NPC presents a very variable incidence according to areas of the world. It is a relatively rare cancer except in China, Southeast Asia and in the Maghreb, where the incidence rates are high. In these areas, the NPC is a major problem of public health. The studies suggested the existence of a great number of factors of environmental risks including food practices and domestic and professional environment. 650 cases and 639 witnesses were recruited. Data extracted from a questionnaire comprise 986 samples and 61 characters. These samples represent a population of individuals whose age exceeds 35 years old.

In the 61 characters, the first variable represents the NPC and the other ones are the explicative variables with the following interpretation: age of interview for control individuals and age at cancer for cases 2, sex 3, instruction 4, professional category 5, lodging ch. and ad. 6 7, parents consanguinity 8, otitis 9, pharyngitis 10, cold 11, asthma 12, eczema 13, allergy 14, chemical manure and pesticide 15, chemical products 16, smoke 17, dust 18, formaldehyde 19, alcohol 20, tabac 21, neffa 22, cannabis 23, housing type ch. and ad. 24 25, separated beds ch. and ad. 26 27, animal in the house ch. et ad. 28 29, kitchen ventilation ch. et ad. 30 31, house ventilation ch. and ad. 32 33, incense ch. and ad. 34 35, kanoun and tabouna ch. et ad. 36 37, wood fire ch. et ad. 38 39, brest feeding and age of weaning and way of weaning 40 41 42, contact with adult saliva 43, traditional childhood treatments 44, hot pepper 45, smen and fat ch. and ad. 46 47, vegetables and fruits ch. and ad. 48 49, house made harrissa ch and ad. 50 51, industrial harrissa ch. ad. 52 53, house made proteins ch. and ad. 54 55,

industrial proteins ch. and ad. 56 57, industrial canned vegetables ch. and ad. 58 59, house made canned vegetables ch. and ad. 60 61. {ch.=childhood and ad=adulthood.}

We apply the version of the binary topological map to the data described above with an architecture of (10 × 10) cells. the training of this map provides for each cell a referent w_c with two codings: disjunctive and additive. Fig.1 represents the distribution of the population on the map (the number of individuals in each neuron). Fig.2 represents the distribution by distinguishing the cases cancer and not-cancer. With the help of the obtained topology, we can already carry out some analysis on the distribution of the individuals. Indeed, there are 5 empty neurons, i.e neurons which do not include any individual. These neurons represent the property of smoothing, one of the strong properties which characterize the self-organizing map. The neurons are mixed, that means that some individuals not having cancer, even if they are minority in a neuron, can have the same characteristics as individuals having cancer!

44	31	21	25	21	11	14	15	17	15
18	11	5	0	14	13	3	0	2	8
33	5	5	11	9	1	6	12	5	16
37	14	13	4	4	6	7	4	6	0
11	16	5	4	3	12	8	0	10	11
14	8	3	13	8	15	4	6	4	0
22	5	5	7	5	9	5	4	6	5
13	14	6	7	2	10	6	7	4	8
8	10	5	8	3	13	4	8	1	16
37	6	12	7	6	21	9	7	9	5

Fig. 1. Topological map (10 × 10) with neuron's cardinalities

To optimize the number of neurons obtained in the map, we applied the method of K-means [9] to the referents with various partitions and we chose the one which minimizes the Davis Bouldin index [2]. The optimal number of classes found is equal to 6 (Fig.3). From the optimal index indicated in this last figure, we devised the map on 6 great classes (Fig.4). In this figure, we noted the following statistics: Class 1: the blue area which gathers 13% of the population including 67% having cancer. Class 2: The pink area, gathering 13% of the population including 53% having cancer. Class 3: the green area representing 25% of the population including 65% of Not-cancer. Class 4: red area, representing 18% including 60% having cancer. Class 5 and 6 (31% of the population): yellow and

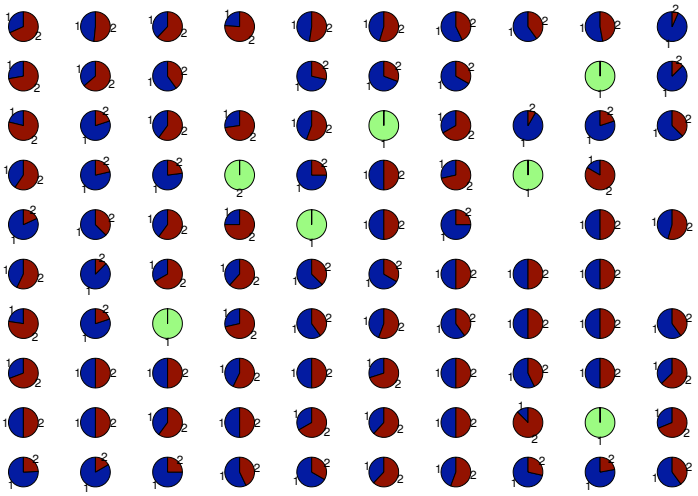


Fig. 2. Distribution of cases Cancer:1 and Not-Cancer:2.

orange, represent "confusion" areas (50% of Cancer and 50% of Not-Cancer), which we cannot interpret!

Regarding these results, two classes draw our attention: Class 1 and Class 3 according to both: 1) the abnormal proportion of individuals reached of cancer taking into consideration 50 % in the full population, and 2) the significant number of individuals in these classes. To characterize the profiles of individuals in these classes, we studied the distribution of the explanatory variables. The divergence of Kullback-Leibler [3] noted $KL(p||q)$ (i.e., $-\sum_x p(x)\ln(q(x)/p(x))$ (in the discrete case) is classically used to measure dissimilarity between the origin distribution $p(X)$ of a variable and that observed in the considered class, $q(X)$. Also, we classified the 60 explanatory variables in each class in descending order of KL. The main principle is to say that a distribution of one variable in a given class is more discriminant than its distribution in the full population.

In class 1 for which there is 67% of individuals reached of the NPC, We observed that 3 variables (50, 45 and 51) have high values of KL (0.75, 0.59 and 0.52 respectively) contrary to the others (<0.25). They are associated to the consumption of spiced food and harrissa in childhood and adulthood. More precisely, we observed in this class that 60 % of the individuals consumed harrissa in childhood whereas this rate is only 20% in the full population.

In class 3 characterized by 65% of individuals which are not reached of the NPC, we observed that variable 26 has a high value of KL (0.8) followed by 32, 30 and 24 (0.38, 0.32, 0.31) contrary to the others (<0.25). Variable 26 is associated

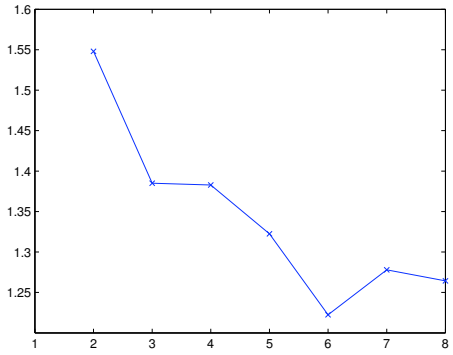


Fig. 3. Davis Bouldin index for choice of optimal number of clusters.

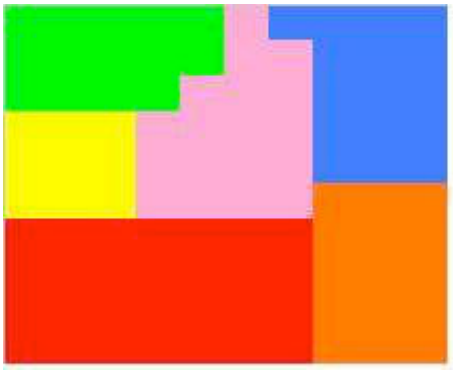


Fig. 4. Division of the Topological map on 6 classes.

to separated beds in childhood, the others relate to the ventilation of the house, the kitchen and the category of housing in childhood. One observes in this class that 90% of the individuals had separated beds in childhood whereas this rate is only 40% in the studied population. In the same way, 22% lived in a quite ventilated house during childhood whereas this rate is only 2% in the studied population. After that, we were interested to extract some rules for explaining the NPC. For that, we carried out a decision tree based on ID3 algorithm [11] with "Gini" function [12]. For example, for Class 1 (blue area), the decision tree is given in Fig.5. where we can remark that the relationship between cancer and the consumption of pimentos is also confirmed by decision tree but in details in the form of rules.
The red and green bars in Fig.5. represent Cancer and Not-Cancer, respectively.
Node 1: represents all individuals in Class 1 (67% of Cancer)

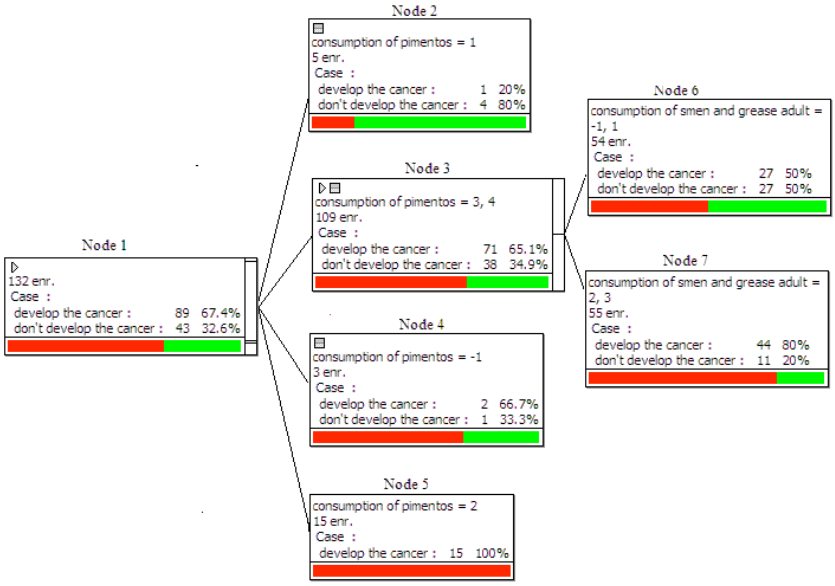


Fig. 5. Decision tree for Class 1.

Node 2: represents all individuals in Class 1 which consume few of pimentos (the proportion of NPC is low)
Node 3: represents all individuals in Class 1 which consume much of pimentos (the proportion of NPC is important)
Node 4: is not considered. It represents the missing values in the variable.
Node 5: represents all individuals in Class 1 which fairly consume pimentos (The proportion of NPC is high).

We can remark two other developed nodes (6 and 7) from Node 3, representing The consumption of "smen" and grease in adulthood. However this separation represents a low discrimination. Subsequently, we can give the following rule :
NPC can probably be explained by the high consumption of pimentos and possibly by the consumption of high consumption of smen and grease in adulthood.

Finally, this type of analysis informs us about the existence of distinct statistical profiles in the studied population. The analysis of the profiles in each class allowed us to identify factors correlated with cancer (or the absence of cancer). By this type of analysis we could draw up semantic statistical profiles of the various categories of individuals reached or not of cancer and thus extract the explanatory variables of each profile in the form of rules.

6 Conclusion

In this paper, we presented a data mining system based on a binary topological map model and decision tree algorithm. This system aims to extract profiles of Nasopharyngeal carcinoma (NPC) using the data of an epidemiologic study. This system uses a binary coding to code the qualitative data and the missing values. The binary topological map allowed us to find homogeneous clusters (with different cases: Cancer, Not-Cancer) and to extract the explanatory variables from each calculated profile. The analysis of the result shows the possibility of the resulting map : in topology preserving order and in the reconstitution of binary components in clustering. Thus, the topological map provides to the epidemiologists a visualization tool of multidimensional qualitative data. This tool allows us to split general profile of the population having the NPC on a set of different profiles. each profile is then more explained by some rules extracted with the help of decision tree algorithm.

References

1. Anouar, F. Badran, F. S. (1998). Probabilistic self-organizing map and radial basis function. Neurocomputing 20, 8396.
2. Bezdek, J.C. and N.R. Pal: Some new indexes of cluster validity. IEEE Transactions on Systems, Man and Cybernetics, 28(3), (1998), pp. 301-315.
3. Bishop, C. (2006). Pattern recognition and machine learning. Berlin: Springer.
4. Breiman L., J.H. Friedman, R.A. Olshen, C.J. Stone, (1984), Classification and Regression Trees, Ed: Wadsworth and Brooks. Montery, California.
5. Diday, E. et C.Simon. (1976). An Introduction to Symbolic Data. New York: Springer.
6. Kohonen, T. (1994). Self-Organizing Map. Berlin: Springer.
7. Lebbah, M., S. Thiria, et F. Badran (2000). Topological map for binary data. In European Symposium on Artificial Neural Networks (ESANN00), pp. 267 - 272.
8. Leich, F., F. Weingessel, et A. Dimitriadou (1998). Competitive learning for binary data. In International Conference on Articial Neural Networks mining (ICANN98), pp. 779 - 784.
9. McQueen, J. Some methods for classification and analysis of multivariate observations. In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 1:281-297, 1967.
10. Marchetti, F. (1989). Contribution a la classification de donnes binaires et qualitatives. Thèse de doctorat, Universit de Metz.
11. Quinlan, J.R., Induction of Decision Trees, Machine Learning (1), 82 - 106, 1986.
12. Zighed, A., Rakotomalala, R. Graphes d'induction - Apprentissage et Data mining . Herms Paris. (2000).

An incremental distributed algorithm for a partial Grundy coloring of graphs

Lyes Dekar[1], Brice Effantin[2], Hamamache Kheddouci[2],

[1] Université de Lyon, Lyon, F-69003, France ;
Laboratoire LIESP, Université Lyon1, IUT A, Département Informatique,
71 rue Peter Fink, F-01000 Bourg en Bresse, France

[2] Université de Lyon, Lyon, F-69003, France ;
Laboratoire LIESP, Université Lyon 1,
Bâtiment Nautibus (ex.710), 843 bd du 11 Novembre 1918,
F-69622 Villeurbanne Cedex, France

{ldekar, beffanti, hkheddou}@bat710.univ-lyon1.fr

Abstract. A coloring of a graph $G=(V,E)$ is a partition of $\{V_1,V_2...V_k\}$ of V into k independent sets called color classes. A vertex $v \in V_i$ is called a Grundy vertex if it is adjacent to at least one vertex in color class V_j, for every $j<i$. In the partial Grundy coloring, every color class contains at least one Grundy vertex. Such a coloring gives a partitioning of the graph into clusters for which every cluster has a clusterhead (the Grundy vertex) adjacent to some other clusters. Such a decomposition is very interesting for large distributed systems and networks. In this paper, we propose a distributed algorithm to maintain the partial Grundy coloring of any graph G when an edge is added.

Keywords: distributed algorithm, partial Grundy coloring, vertex coloring.

1 Introduction

We consider graphs without loops or multiple edges. Let $G=(V,E)$ be a graph where $|V|=n$ and $|E|=m$. We define a *k-coloring* of G as a function c defined on $V(G)$ into a set of colors $C=\{1,2,...,k\}$ such that for each vertex $x \in V(G)$, $c(x) \in C$. A *proper k-coloring* is a k-coloring verifying the condition $c(x) \neq c(y)$ for any pair of adjacent vertices $x,y \in V(G)$. Then, we define a *Grundy vertex* as a vertex v of G, adjacent to vertices colored by every color 1 to $c(v)-1$. A *Grundy coloring* is then defined as a proper coloring where every vertex is a Grundy vertex. In 1979, Christen and Selkow [3] introduced a parameter, the *Grundy number*, which maximizes the number of colors such that a graph admits a Grundy coloring. This parameter was very studied for different classes of graphs (trees [11, 17], power graphs [10], cartesian product of graphs [5],…). It was also compared to several other parameters (ochromatic number [7], achromatic number [11], Grundy number of the complement graph [4],…). Many algorithms were also proposed to determine the Grundy coloring of a graph ([12, 13]).
More recently, Erdös et al. [8] presented the concept of *partial Grundy coloring*, which is a refinement of a Grundy coloring. A *partial Grundy k-coloring* is a proper k-coloring where any color has at least one Grundy vertex. Then the *partial Grundy number* of a graph G, denoted $\delta\Gamma(G)$, is the maximum integer k such that G admits a partial Grundy k-

coloring. In [8], Erdös et al. related the partial Grundy number to the upper ochromatic number. More recently, Shi et al. [14] showed that the problem of determining the partial Grundy number of a graph is NP-complete even for chordal graphs and bipartite graphs. They also proposed a linear-time algorithm to determine the partial Grundy number of trees and graphs with girth larger than 8.

We can remark that a minimal coloring is by definition a Grundy coloring because the color of each vertex is the smallest color not appearing in its neighborhood. A minimal coloring can be provided in a polynomial time as appears in [12]. It is clear that a Grundy coloring is also a partial Grundy coloring, since each color class contains a Grundy vertex. Therefore, in our paper we do not provide only a partial Grundy coloring by using a minimal number of colors (obtained with a minimal coloring), but we try to maximize as possible the number of colors in the graph. It is very important to note that the coloring constraint in the partial Grundy coloring is not only on the neighborhood of each vertex, but on the entire graph, contrary to the Grundy coloring. Indeed, the color of a vertex depends on the presence of a Grundy vertex for its color. Then, the determination of a partial Grundy coloring becomes more difficult to compute.

In this paper, we propose an incremental distributed algorithm to maintain a partial Grundy coloring of a graph when edges are added to this one. Our algorithm tries also to maximize (as possible) the number of colors in the graph. A general motivation of our work is that coloring problems are very interesting tools to model conflicts in networks for example. Moreover, a vertex coloring of a graph gives a particular partitioning (or clustering) of this graph. Thus, the studied coloring allows us to partition the graph into color classes and in each class, we determine a clusterhead able to join directly some other classes (those with less colors). Such a clustering can have many interests for example in the routing of information where it is very useful to join particular sets of vertices or to limit the length of a path. Furthermore, such an algorithm can be used to determine a partial Grundy coloring for any graph G (a problem for which no polynomial solution exists). The principle of this solution would be to initialize the graph by coloring a spanning tree T of *G* (by using a distributed version of the algorithm proposed in [14] for instance), and to use a second step to add the edges of G\T. The determination of a spanning tree is a well-known problem for which a lot of polynomial algorithms exist [1, 2, 9].

We first start by some definitions and notations used in the remainder of the paper. For a vertex v, we denote its neighborhood by $N(v)$ (note that the set of colors of $N(v)$ is denoted $N_c(v)$). We let $d(v)=|N(v)|$, the number of neighbors of vertex v, or its *degree*, and we let $\Delta=\max\{d(i)|i\in V\}$. In a partial Grundy coloring, we denote the Grundy vertex in each color class i by g_i. Note that, if several vertices can be Grundy vertices for color i, only one is considered. The set of Grundy vertices $\{g_1,g_2,\ldots,g_k\}$ forms the Grundy sequence $(g_1,g_2,\ldots,g_k)$ of the graph. A Grundy vertex g_i is said *satisfied* if for any $q\in[1, i-1]$, there exists at least one vertex $x\in N(g_i)$ such that $c(x)=q$. Then, in our algorithm, vertices can have different states. Thus, we define a data, specific to each vertex, denoted *status*, to precise the state of each vertex. A vertex v can have three different states:

Grundy: a Grundy vertex is adjacent to all colors from 1 to $c(v)-1$. Note that $v.status=Grundy$ means $g_{c(v)}\equiv v$.

satisfaction: a satisfaction vertex is used by a Grundy vertex to be satisfied. It is so a neighbor of a Grundy vertex. Note that if several neighbors of a Grundy vertex has the same color, only one will be a satisfaction vertex. Thus, around a Grundy vertex, we find only one satisfaction vertex for each color $1,2,\ldots,c(v)-1$.

normal: a vertex is normal if it is neither a Grundy nor a satisfaction vertex.

The remainder of the paper is organized as follows. In Section 2 we present a first procedure used in our distributed algorithm, to evaluate the color for which a vertex can become a Grundy vertex. This procedure is important to maximize the total number of colors. Then, in Sections 3 we propose a distributed approach to determine a partial Grundy coloring of a graph.

2 The possible Grundy color set

In this section, we propose a procedure that allows determining the possible colors for which a vertex v can become a Grundy vertex. This step is important in the proposed algorithm for the partial Grundy coloring since it allows a vertex to determine if it can be a Grundy vertex for a color.

Let G be a graph with a partial Grundy k-coloring. Each vertex x of the graph can easily maintain a list of colors it can get, defined by $L(x)=\{1,2,\ldots,k\}\backslash c(w)$ for any $w\in N(x)$ such that w.status=Grundy or w.status=satisfaction. In other terms, L(x) contains all the colors not appearing on a Grundy or satisfaction neighbor of x. Let v be the vertex of G we want to evaluate. We define by P(v), all the possible colors that v can take to become a Grundy vertex. Note that $\forall c\in P(v)$, we have $c \leq d(v)+1$. To become a Grundy vertex, v must place each color c, with $c<c(v)$, on one of its neighbors. In order to find P(v), we generate a bipartite graph $H=(V_1,V_2,E')$ where the two vertices sets are $V_1=\{x_1,x_2,\ldots,x_{|L|}\}$ with $L=\bigcup_{x\in N(v)} L(x)$ and $V_2=\{y_1,y_2,\ldots,y_{|N(v)|}\}=N(v)$. Thus, V_2 is the set of all the neighbors of v, and V1 is the set of all the colors that can get the vertices of V2. These two sets are initially colored as follows: $c(x_i)=c$ such that $c\in L$ and $c \neq c(x_j)$ for $1 \leq i \neq j \leq |L|$, and for any $y_i\in V_2$ we have $c(y_i)=\Delta+2$. Then, an edge $(x_i,y_i)\in E'$ exists if $c(x_i)\in L(y_i)$. The Figure 1 presents an example of construction of a bipartite graph H.

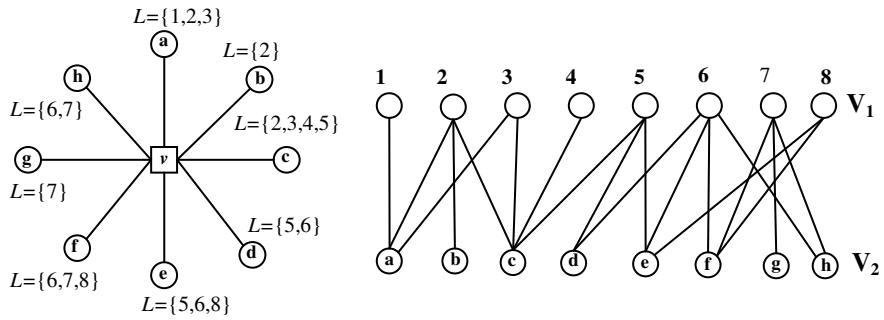


Fig. 1. For a vertex v and its neighborhood (a) we compute the bipartite graph H (b)

Procedure 1: *Possible_Grundy_color (v)*
BEGIN
Let $H=(V_1,V_2,E')$ be the bipartite graph defined by $V_1=\{x_1, x_2,\ldots, x_{|L|}\}$ where $L=\bigcup_{x\in N(v)} L(x)$, $V_2=\{y_1,y_2,\ldots,y_{|N(v)|}\}=N(v)$, where each node x_i has a different color of L and for any $y_i\in V_2$ we have $c(y_i)=\Delta+2$, and $(x_i,y_i)\in E'$ exists if $c(x_i)\in L(y_i)$.

While $V_1 \neq \varnothing$ **do**
Let x be the vertex of V_1 such that $c(x)=min\{c(u)|u \in V_1\}$.
While $x \in V_1$ **do**
Let y be a vertex of V_2 such that $d(y)=min\{d(u)|u \in N(x)$ *and* $c(u)>c(x)\}$.
If y exists **then** $c(y)=c(x)$. **Endif**
Remove x from $V_1(H)$.
If y exists and $N(y) \neq \varnothing$ **then** Let x be the vertex such that $c(x)=min\{c(u)|u \in N(y)\}$.
Endif
Endwhile
Endwhile
Let $q=min\{c|c \geq 1$ *and for any* y_i, $c(y_i) \neq c\}$.
Let Q be the set of colors $\{1,2,\ldots,q-1\} \backslash c(y_i)$ for any y_i such that $y_i.status=Grundy$ or $y_i.status=satisfaction$.
Return Q.
END

Then, we provide a procedure to find an optimal repartition of consecutive colors on the neighbors of v, represented by the vertices of V_2. The aim of this procedure is to place a maximum number of consecutive colors of V_1 on vertices of V_2 starting from the lowest one to determine the colors for which v can be a Grundy vertex. This procedure is called *Possible_Grundy_color (v)*. A color is put on a vertex once and only once, and is removed from H after the affectation. Then, two vertices of V_2 cannot have the same color. This process continues until $V_1=\varnothing$. After this step, some colors may be not affected. Let q be the smallest color not on a vertex of V_2. We can deduce that all the colors 1 to $q-1$ appear on V_2. To avoid perturbing the existing coloring (in particular Grundy vertices), v cannot be a Grundy vertex for colors appearing on Grundy or satisfaction neighbors of v. But v can be a Grundy vertex of all other colors from 1 to q.

Procedure 1: Possible_Grundy_color (v)
BEGIN
Let $H=(V_1,V_2,E')$ be the bipartite graph defined by $V_1=\{x_1, x_2,\ldots, x_{|L|}\}$ where $L=\bigcup_{x \in N(v)} L(x)$, $V_2=\{y_1,y_2,\ldots,y_{|N(v)|}\}=N(v)$, where each vertex x_i has a different color of L and for any $y_i \in V_2$ we have $c(y_i)=\Delta+2$, and $(x_i,y_i) \in E'$ exists if $c(x_i) \in L(y_i)$.
While $V_1 \neq \varnothing$ **do**
Let x be the vertex of V_1 such that $c(x)=min\{c(u)|u \in V_1\}$.
While $x \in V_1$ **do**
Let y be a vertex of V_2 such that $d(y)=min\{d(u)|u \in N(x)$ and $c(u)>c(x)\}$.
If y exists **then** $c(y)=c(x)$. **Endif**
Remove x from $V_1(H)$.
If y exists and $N(y) \neq \varnothing$ **then** Let x be the vertex such that
$c(x)=min\{c(u)|u \in N(y)\}$. **Endif**
Endwhile
Endwhile
Let $q=min\{c|c \geq 1$ and for any y_i, $c(y_i) \neq c\}$.
If $q>c(v)$ and v.status=satisfaction **then** $Q=c(v)$
Else Let Q be the set of colors $\{1,2,\ldots,q\} \backslash c(y_i)$ for any y_i such that y_i.status=Grundy or y_i.status=satisfaction.
EndIf
Return Q.
END

In the following, we prove that the set of colors returned by Procedure 1 contains colors for which the vertex v can be a Grundy vertex.

Lemma 1 Procedure 1 determines a set of colors for which the vertex v can be a Grundy vertex with a $O(\Delta)$ complexity.

Proof. Let $H=(V_1,V_2,E')$ be the bipartite graph defined in Procedure 1. Then this procedure affects colors to vertices $y_i \in V_2(H)$. Let $q=\min\{c | c \geq 1$ and for any y_i, $c(y_i) \neq c\}$. Then, vertex v could be a Grundy vertex for any color 1,2,…,q. However, to keep the coloring proper, it cannot take a color of neighbors which are Grundy vertices or satisfaction vertices (since they cannot change their color). So the vertex v can be a Grundy vertex for any color of the set returned by Procedure 1. Moreover, each color of V_1 is assigned to a vertex of V_2 only once. Since the number of colors is equal to $|V_1|=\Delta$ then the complexity of Procedure 1 is $O(\Delta)$.
We can easily associate to this procedure, a second $O(\Delta)$ procedure, denoted *Put_Grundy(v,c)*, used to color the vertex v and some of its neighbors according to the colors affectation performed by Procedure 1, in order to make v a Grundy vertex for color c and its neighbors satisfaction vertices for colors 1,…, $c-1$.

3 Distributed approach for a partial Grundy coloring

3.1 Existence of a partial Grundy coloring

In this section we study the following question: if G is a graph with a partial Grundy coloring, can such a coloring be maintained when new edges are added to G?
Then, we prove that, from any graph G admitting a partial Grundy coloring, if an edge is added, then a partial Grundy coloring can always be found in the resulting graph.
If an edge is added between two vertices with different colors, every vertex does not need to change neither its color nor its state.

Lemma 2 *Let $G=(V,E)$ be a graph with a partial Grundy coloring. Let x and y be two non adjacent vertices of G where $c(x) \neq c(y)$. Let $G'=(V,E')$ be a graph such that $E'=E \cup \{(x,y)\}$. Then, the graph G' has a partial Grundy coloring.*

Proof. As the edge is added between two vertices with different colors then the coloring of G' is proper. Since no vertex in the graph changes its color, then every Grundy vertex in G is a Grundy vertex in G'. The coloring of G' is a partial Grundy coloring.

A conflict appears only if a new edge joins two vertices with the same color. Then, if an endpoint of a new edge is a normal vertex, only this vertex needs to change its color.

Lemma 3 *Let $G=(V,E)$ be a graph with a partial Grundy coloring. Let x and y be two non adjacent vertices of G such that $c(x)=c(y)$ and x is a normal vertex. Let $G'=(V,E')$ be a graph such that $E'=E \cup \{(x,y)\}$.Then, the graph G' admits a partial Grundy coloring.*
Proof. Since $c(x)=c(y)$ then the coloring of G' is not proper. Since the vertex x is a normal vertex, it can change its color without influencing the Grundy vertices of the graph. The vertex x takes the minimum color $c \notin N_c(x)$. The Grundy vertices of the graph G stays Grundy vertices for G'. Then, the coloring of G' is a partial Grundy coloring.

Grundy and satisfaction vertices have less possibilities to change their color. However, the removal of at most one color provides a partial Grundy coloring of the graph.

Lemma 4 *Let G=(V,E) be a graph with a partial Grundy k-coloring. Suppose that a color j, with $1 \leq j \leq k$ is removed from the coloring of G. Then, there exists a partial Grundy k'-coloring of G where $k' \geq k-1$.*

Proof. Let $(g_1,g_2,\ldots,g_k)$ be a Grundy sequence of G. Let S be the set of vertices colored with color j. If we put $c(s)=\varnothing$ for any $s \in S$, then the color j is removed and is not represented in the Grundy sequence. Thus, each vertex g_i, with $i>j$, is not any more a Grundy vertex. By decreasing the coloring of every vertex v of G such that $c(v)>j$, the subgraph G\S is colored with k-1 colors. Since any Grundy vertex g_i could be a Grundy vertex for colors 1 to $i-1$, a Grundy sequence $(g_1,g_2,\ldots,g_{k-1})$ is deduced. To complete the coloring of G, each vertex $s \in S$ is colored by $c(s)=\min\{q | q \notin N_c(s)\}$. Thus, if a vertex s is adjacent to the $k-1$ existing colors, it introduces a new color and it becomes its Grundy vertex. Therefore $k' \geq k-1$.

Lemma 5 *Let G=(V,E) be a graph with a partial Grundy k-coloring. Let x and y be two non adjacent vertices such that x and y are not normal vertices and c(x)=c(y). Let G'=(V,E') be a graph such that $E'=E \cup \{(x,y)\}$. Then, the graph G' admits a partial Grundy coloring.*

Proof. The new edge makes the coloring not proper. Since x and y are not normal vertices, they cannot change their color. Therefore, if we remove the color $c(x)$ from the graph, then Lemma 4 shows that there exists a partial Grundy coloring for G'.

Thus, the following theorem proves that from any graph G, a partial Grundy coloring can be found for any graph $G \cup \{(i,j)\}$, where $i,j \in V(G)$.

Theorem 6 *Let G=(V,E) be a graph with a partial Grundy k-coloring. Let x and y be two non adjacent vertices of G. Let G'=(V,E') be a graph such that $E'=E \cup \{(x,y)\}$. Then, G' admits a partial Grundy k'-coloring with $k' \geq \max\{\chi(G),k-1\}$.*

Proof. If $c(x) \neq c(y)$ then by Lemma 2 the partial Grundy coloring is maintained. Otherwise, if x is a normal vertex then by Lemma 3 we can always find a partial Grundy k-coloring. In the other cases, Lemma 5 shows that a partial Grundy k'-coloring with $k' \geq k-1$ can be determined. Moreover, any change of color maintains a proper coloring. By definition, if a coloring is proper, it has at least $\chi(G)$ colors. Thus, G' admits a partial Grundy k'-coloring where $k' \geq \max\{\chi(G),k-1\}$.

3.2 Distributed algorithm

In a distributed system, a vertex exchanges information only with its neighborhood. Every vertex has a set of local variables to determine the local state of the vertex. The state of the entire system is the union of the local states of all the vertices in the system. Distributed algorithms are a very attractive topic for a lot of fields and several graph problems arise naturally in distributed systems. For example, distributed algorithms for finding spanning trees, matchings, independent sets or particular colorings have been studied [6,15,16]. Such algorithms are so very interesting and efficient for dynamic networks.

We propose in this section a distributed algorithm to maintain a partial Grundy coloring when an edge is added to the graph. Thus, we suppose that the graph is initialized by a partial Grundy coloring. Since each vertex is autonomous, then any event or change in its neighborhood can imply a local processing that can be extended to other vertices in the graph. The edge adding is a change for which the two endvertices can react. This reaction consists to execute a set of actions according to the state of the vertex. So, both the vertices check if the edge causes a color conflict or if it can use the new edge to introduce a new color in the graph in order to increase the total number of colors. In the case where a color conflict is detected then the vertex tries to solve the problem locally by changing its state. This change can make the coloring not partial Grundy because Grundy vertices become not satisfied and thus the Grundy sequence not complete. Since the vertices are independent, Grundy vertices are able to detect that the coloring is not partial Grundy since they can verify if they are Grundy vertices. If they are not, they start a set of procedures to re-establish a partial Grundy coloring. Some local variables are needed to the execution of our distributed algorithm.

x.status: the status of the vertex *x* (*Grundy*, *satisfaction*, *Normal*)
x.t[j]: defines the sequence number of the vertex j.
x.k: defines the number of colors of the graph.
x.request: represents the set of colors for which a Grundy vertex is searched.
x.response: represents the set of colors for which at least one Grundy vertex has been found.
x.search: similar to *x.request* and defined for the needs of our algorithm.
x.wait: indicates if a vertex is in a waiting state (1: waiting state, 0: not waiting state).
x.del: indicates the color to delete from the graph to re-establish a partial Grundy coloring.

Some actions of our algorithm consist to broadcast a message to all the vertices in the graph. The reception of a message by a vertex causes the execution of a procedure linked to this message. To avoid the multiple execution of a procedure by a vertex, we use a sequence number. We assume that each vertex maintains a sequence number of each vertex in the graph. A vertex increases by one its sequence number and joins it every time it initiates a message. A vertex that receives this message executes the procedure only if the sequence number associated to the message sender is different to the sequence number received in the message (this means the vertex considers this message once).

3.2.1 General working of our algorithm

Our algorithm is composed of several procedures. The first one is called *Processing()* and is performed continuously on each vertex of the graph. Then, if a new edge causes a perturbation in the system, the concerned vertices will detect it and performs the adequate processing according to their state. *Processing ()* treats all the cases that can happens when an edge is added to a graph. So, if a vertex is not colored and no one of its neighbors is in a waiting state then it takes the smallest color not appearing in its neighborhood. If an edge is added between a normal vertex and any other vertex with the same color, then the normal vertex change its color to maintain a proper coloring. In all these cases, *Maximize_colors ()* procedure is called in order to increase the number of colors in the graph (if possible). This procedure uses the *Possible_GrundyColor(x)* procedure provided in Section 2, in order to determine if the vertex *x* can be a Grundy vertex for the color *k+1*, where *k* is the number of colors in the graph. In this case, the vertex *x* takes the color k+1 and puts the colors 1,…,*k* in its neighborhood. Then, it

informs all the other vertices to update their knowledge about the number of colors in the graph. The updating operation is performed by sending an *Update* message in the graph.
If a satisfaction vertex u detects a color conflict when an edge is added (the new edge causes a non proper coloring), then it changes its state to become a normal vertex. This change of state can make some Grundy vertices not satisfied and then not Grundy. To avoid decreasing immediately the number of colors, these vertices will try to put the needed colors (to stay Grundy) on normal neighbors or will try to find new Grundy vertices for their colors by sending *Search_Substitute* messages in the graph. The senders and the receivers of these messages become in a waiting state for time $2diam(G)+1$. We note that diam(G) (the diameter of the graph G) is given and known before executing the algorithm. The diameter of a graph can only decrease when edges are added. When a vertex receives a *Search_Substitute* message, it checks if it can be a Grundy vertex for the color specified in the message by using the *Possible_GrundyColor* procedure. If this is possible, then it becomes Grundy for this color and sends a *Substitute_Found* message to inform the other vertices that a new Grundy vertex is found for this color. Otherwise, it propagates the *Search_Substitute* message. At the end of the waiting delay, if no substitute was found for at least one Grundy vertex v, then a color will be removed from the coloring. Since many vertices can declare themselves as substitutes for the same Grundy vertex v then only one is selected. Therefore, we can observe that our strategy is to maximize as possible the number of colors in the graph by: in one hand avoiding as possible to decrease the number of colors, and in the other hand trying to increase it at each edge adding.

3.2.2 Procedures used during the execution of our algorithm

In this subsection, we describe all the procedures used during the execution of our distributed algorithm. The main procedure of our algorithm is Processing() which initiates the appropriate actions according to the graph changes (adding of edges and not proper coloring).

Procedure 2: ***Processing ()***
BEGIN
If $c(x)=Null$ and $\forall z\in N(x)$, $z.wait=0$ **then**
  $c(x)=\min\{l\geq 1 | \forall j\in N(x), c(j)\neq l\}$.
  Maximize_colors ().
Endif
If $x.status=normal$ and $c(x)\neq Null$ **then**
  If $c(x)\in N_c(x)$ **then** $c(x)=\min\{l\geq 1 | \forall j\in N(x), c(j)\neq l\}$. **Endif**
  Maximize_colors ().
Endif.
If $x.status=satisfaction$ and $\exists y\in N(x) | y.status=satisfaction$ or *Grundy* and $c(y)=c(x)$ **then**
  x.status= *normal*.
Endif.
If $x.status=Grundy$ and $\exists q\in\{1,2,\ldots,c(x)-1\}$ such that $\exists y\in N(x)$ / $c(y)=q$ and $y.status=satisfaction$ **then**
  If $\exists z\in N(x) | z.status=normal$ and $q\in L(z)$ **then** (we recall that $L(z)$ is the set of colors that z can take)
    $c(z)=q$.
  Else
    $x.t[x]=x.t[x]+1$.
    $x.request=x.request\cup\{c(x)\}$.

x.wait=1.
Send to each $z \in N(x)$: *Search_Substitute(x, j.t[x], c(x), q)*.
Start *Waiting ()*.
Endif
Endif
END

The following procedure is used to check if a vertex can take a new color ($k+1$) to increase the number of colors of the graph. If it can do it then it declares itself a Grundy vertex for this color and informs the other vertices in the graph.

Procedure 3: ***Maximize_Colors ()***
BEGIN
If *x.status=normal* and $x.k+1 \in$ *Possible_GrundyColor(x)* **then**
c(x)=x.k+1.
x.k=x.k+1.
x.status=Grundy.
Put_Grundy(x,k+1).
x.t[x]=x.t[x]+1.
Send to each $v \in N(x)$: *Update (x.k, x.t[x], x)*.
Endif
END

If a vertex takes a new color in the graph then it informs all the other vertices by sending an *Update (c, t_0, i)* message. Every vertex *x* that receives this message (initiated by vertex *i*) updates its number of colors. If many Grundy vertices have this color then only one among them stays Grundy. The selected vertex, to stay Grundy for this new color *c*, is the vertex with the smallest value in the set returned by *Possible_GrundyColor(x)* procedure. The other Grundy vertices (and their satisfaction vertices) for this color become normal.

Procedure 4: ***Update (c, t_0, i)***
BEGIN
If $t_0 > x.t[i]$ **then**
$x.t[i]=t_0$.
x.k=c.
If *c(x)=c* and *x.status=Grundy* **then**
If *x>i* **then** Vertex *x* and every satisfaction vertex which satisfies only *x* become normal.
Endif
Endif
Send to each $v \in N(x)$: *Update (c, t_0, i)*.
Endif
END

When a vertex *x* receives a *Search_Substitute(j, t_0, c, q)* message (initiated by vertex *j*), it checks if it can be the Grundy vertex for the color *c*. In order to be a Grundy vertex for *c*, the vertex must be a normal or satisfaction vertex and the color *c* must belong to the set of Grundy colors computed by the *Possible_GrundyColor* procedure. Note that a satisfaction vertex can be Grundy for its color only. If the two conditions are satisfied then *x* becomes a new Grundy vertex for the color *c* and it informs the remaining vertices by sending a *Substitute_Found* message. Otherwise it propagates the received message and notes that color *q* will be the removed color if no Grundy vertex is found for color *c*.

Procedure 5: ***Search_Substitute (j, t_0, c, q)***
BEGIN
If $t_0 > x.t[j]$ **then**
$x.t[j]=t_0$
$x.del=q$.
if $|x.request|=0$ **then**
waiting().
$x.wait=1$.
Endif
$x.request=x.request \cup \{c\}$.
If $c \in Possible_GrundyColor(x)$ and *x.status=normal or satisfaction* **then**
x.status= Grundy.
c(x)= c.
Put_Grundy(x,c).
$x.t[x]=x.t[x]+1$.
$x.response=x.response \cup \{c\}$
Send to each $z \in N(x)$: *Substitute_Found (x, x.t[x], c, j).*
Else
Send to each $z \in N(x)$: *Search_Substitute (j, t_0, c, q).*
Endif
Endif
END

A vertex receives a message for any color without Grundy vertices. Thus, if a new Grundy vertex is found for these colors, a new message is received. Thus, after a delay $2diam(G)+1$, a vertex received a first message from any ex-Grundy vertices and a second message from any new Grundy vertices. If some colors have no Grundy vertices, the coloring must be modified and a color will be removed.

Procedure 6: ***Waiting ()***
BEGIN
After a time $2diam(G)+1$ **do**
If $|x.response| < |x.request|$ **then**
$x.k=x.k-1$.
If $c(x)=x.del$ **then** *c(x)=Null. x.state=normal.* **Endif**
If $c(x)>x.del$ **then** $c(x)=c(x)-1$. **Endif**
$x.request=\emptyset$.
$x.response=\emptyset$.
$x.wait=0$.
Endif
Enddo
END

When the Grundy vertex *j* to replace receives a *Substitute_Found (p, t_0, c, j)* message, it puts its status and the status of its satisfaction vertices to normal. Since several substitutes can appear, only the substitute with the smallest value in the set returned by *Possible_GrundyColor(x)* procedure is selected. This allows the others to be Grundy vertices for larger colors. These substitutes and their satisfaction vertices become normal.

Procedure 7: ***Substitute_Found*** **(*p, t_0, c, j*)**
BEGIN
If $t_0 > x.t[p]$ **then**
x.response=x.response$\cup\{c\}$.
If (*c(x)=c*, *x.status=Grundy* and *Max{Possible_GrundyColor(x)}>*
Max{Possible_GrundyColor(p)}) or (*x=j*)
then
Vertex *x* and every satisfaction vertex which satisfies only *x* become normal.
If $\exists c' \in$ *x.search* , c' is the closest to *Max{Possible_GrundyColor(x)}* then
x.status= Grundy.
c(x)= c'.
Put_Grundy(x,c').
x.t[x]=x.t[x]+1.
x.response=x.response$\cup\{c'\}$
Send to each $z \in N(x)$: *Substitute_Found (x, x.t[x], c', j).*
Endif
Endif
Send to each $z \in N(q)$: *Substitute_Found (p, t_0, c, j).*
Endif
END

Next, we prove that this algorithm gives a partial Grundy coloring of *G* when an edge is added.

Theorem 7: *Let G be a graph with a partial Grundy coloring. If an edge is added to G, then a partial Grundy coloring is re-established with O(nm) exchanged messages and O(n) color changes.*

Proof. First note that the propagation of a message in all the graph is done in *O(m)* messages. Then, not satisfied Grundy vertices send a *Search_Substitute* message in the graph to find Grundy vertices for their colors. And at most (n-Δ) vertices can become Grundy vertices for one of these colors. Thus, the search of Grundy vertices needs *O(nm)* messages. Since the update message is propagated once for each new color found, therefore, the number of exchanged messages to re-establish a partial Grundy coloring is *O(nm)*. Moreover, each vertex can change its color at most once when a color is removed and once when a Grundy vertex is searched. The determined coloring needs then *O(n)* changes of colors.

Theorem 8: *The time complexity to reach a partial Grundy coloring is O(diam (G)).*

Proof. Each vertex waits for a delay 2*diam(G)*+1 to decide to remove a color (if necessary). Moreover, the time lag between the start of the waiting procedure for two vertices is at most *diam(G)*. Thus, such a coloring is determined in time *O(diam(G))*.

4 Conclusion

In this paper, we propose an incremental distributed algorithm to maintain a partial Grundy coloring of a graph when edges are added to this one. Our algorithm tries also to maximize (as possible) the number of colors in the graph. Our work is motivated by the partitioning provided by the coloring. In addition, the algorithm can be used to computes

a partial Grundy coloring for any graph G (a problem for which no polynomial solution exists).

References

[1] G. Antonoiu and P.K. Srimani, *A self-stabilizing distributed algorithm for minimal spanning tree problem in a symmetric graph*, Computer & Mathematics with Application **35** (10) (1998) 15-23.
[2] M. Bui, F. Butelle and C. Lavault, *A distributed algorithm for constructing a minimum diameter spanning tree*, Journal of Parallel and Distributed Computing **64** (2004) 571-577.
[3] C. A. Christen and S. M. Selkow, *Some perfect coloring properties of graphs*, Journal of Combinatorial Theory **B27** (1979) 49-59.
[4] E.J. Cockayne and A.G. Thommason, *Ordered Colourings of graphs*, Journal of Combinatorial Theory **B32** (1982) 286-292.
[5] B. Effantin and H. Kheddouci, *Grundy number of graphs*, Discussiones Mathematicae Graph Theory (to appear).
[6] B. Effantin and H. Kheddouci, *A Distributed Algorithm for a b-Coloring of a Graph*, International Symposium on Parallel and Distributed Processing and Applications (ISPA 2006), Lecture Notes in Computer Science, vol.4330, pp.430-438 (2006).
[7] P. Erdös, W.R. Hare, S.T. Hedetniemi and R. Laskar, *On the equality of the Grundy and ochromatic numbers of a graph*, Journal of Graph Theory **11** (2) (1987) 157-159.
[8] P. Erdös, S. T. Hedetniemi, R. C. Laskar and G. C. E. Prins. *On the equality of the partial Grundy and upper ochromatic numbers of graphs*. Discrete Mathematics **272** (2003) 53-64.
[9] T. Fujie, *An exact algorithm for the maximum leaf spanning tree problem*, Computers & Operations Research **30** (2003) 1931-1944.
[10] C. Germain and H. Kheddouci, *Grundy numbers of powers of graphs*, Discrete Mathematics (to appear).
[11] S. M. Hedetniemi, S. T. Hedetniemi and T. Beyer, *A linear algorithm for the Grundy (coloring) number of a tree*, Congressus Numerantium **36** (1982) 351-363.
[12] S.T. Hedetniemi, D.P. Jacobs and P.K. Srimani, *Linear time self-stabilizing colorings*, Information Processing Letters **87** (2003) 251-255.
[13] S.-T. Huang, S.-S. Hung and C.-H. Tzeng, *Self-stabilizing coloration in anonymous planar networks*, Information Processing Letters **95** (2005) 307-312.
[14] Z. Shi, W. Goddard, S. T. Hedetniemi, K. Kennedy, R. Laskar and A. MacRae. *An algorithm for partial Grundy number on trees.* Discrete Mathematics **304** (2005) 108-116.
[15] Z. Shi, W. Goddard and S. T. Hedetniemi, An anonymous self-stabilizing algorithm for 1-maximal independent set in trees, Information Processing Letters **91** (2004) 77-83.
[16] P. Šparl and J. Žerovnik, *2-local distributed algorithms for generalized coloring of hexagonal graphs*, Electronic notes in Discrete Mathematics **22** (2005) 321-325.
[17] J.A. Telle, A. Proskurowski, *Algorithms for vertex partitioning problems on partial k-trees*, SIAM Journal on Discrete Mathematics **10**(4) (1997) 529-550.

Optimisation Multi-objectif pour l'Alignement Multiple de Séquences

N. Benlahrache*, S. Meshoul+

* Laboratoire LIRE, équipe PRAI, université Mentouri Constantine,
nbenlahrache@gmail.com,
+ King Saud University, Information Technology Department, Riyadh
meshoul@ksu.edu.sa

Résumé. La bioinformatique est une discipline qui vise le traitement automatique de l'information biologique. L'alignement multiple de séquences (MSA) constitue une tâche fondamentale pour beaucoup d'applications en bioinformatique. Afin d'évaluer un MSA donné, plusieurs fonctions objectif ont été définies moyennant des formulations mathématiques. Cependant toutes les méthodes d'alignement proposées dans la littérature reposent sur des approches d'optimisation mono-objectif et fournissent une seule solution potentielle. Aucune de ces méthodes ne peut prétendre l'optimalité quand il s'agit de fournir, dans tous les cas, des alignements corrects au sens biologique. Dans ce papier, proposons et étudions l'idée de traiter ce problème sous l'angle de l'optimisation multi-objectif. La stratégie que nous proposons est un algorithme génétique multi-objectif qui opère sur un ensemble d'alignements dont l'évolution est assurée par des opérateurs génétiques et conduisant à un ensemble d'alignements de bonne qualité. Les résultats obtenus sont prometteurs et mettent en évidence l'apport de l'optimisation multi-objectif pour la résolution du problème traité.

Mots clés: Optimisation Multi-objectif, Bioinformatique, Alignement Multiple de Séquences, Algorithme génétique Multi-objectif, Dominance au sens de Pareto.

1. Introduction

L'alignement multiple de séquences ou MSA (pour Multiple Sequence Alignment) est un problème fondamental en biologie moléculaire et représente une tâche de base pour beaucoup d'applications en bioinformatique. Ce dernier est un domaine pluridisciplinaire où l'informatique joue un rôle prépondérant. Il est destiné à modéliser, analyser, comparer et simuler l'information biologique incluant séquences, structures, fonctions et phylogénie. MSA est le moyen utilisé par les biologistes pour analyser des séquences d'ADN (nucléiques) ou de protéines (protéiques) afin de déterminer leur degré d'homologie ou de divergence. MSA est utilisé pour la construction d'arbres phylogénétiques et pour l'identification des motifs dans des familles de protéines permettant la prédiction de leur aspect structurel et fonctionnel [1]. Cependant, cette extrême importance de MSA est confrontée à l'extrême difficulté de sa résolution. La raison en est que la recherche d'un alignement de bonne

qualité implique souvent l'exploration d'espaces de recherche très vastes et dont la taille devient de plus en plus critique avec le nombre et les tailles des séquences à aligner. Étant NP-complet, MSA ne peut être résolu par une méthode exacte que pour des séquences de petites tailles et dont le nombre est réduit induisant des espaces de tailles réduites.

Durant la dernière décennie, plus de cinquante méthodes ont été proposées dans ce domaine [2]. Ce nombre risque d'augmenter les prochaines années du fait que le problème reste toujours ouvert et non complètement résolu. Globalement, en excluant les méthodes exactes, les méthodes décrites dans la littérature peuvent être groupées en deux classes : les méthodes progressives et les méthodes itératives telles que Clustal_W [3], SAGA[4], PRRP[5] , DIALIGN [6], T_Coffee[7], MAFFT [8], Muscle[9], Align_M[10], ProbCons[11]… etc. Étant donné un ensemble de séquences, les méthodes progressives se proposent d'effectuer l'alignement en partant des séquences les plus similaires et en ajoutant les séquences restantes graduellement une par une selon un ordre pré-établi. Elles utilisent souvent la programmation dynamique. La simplicité et la rapidité de traitement sont les avantages majeurs de ces méthodes. Cependant, leur nature gloutonne conduit souvent à des solutions sous optimales. Ce qui explique le recours aux méthodes itératives pour gérer la complexité combinatoire du problème. Leur principe de base consiste à produire un alignement initial et à le raffiner itérativement de manière déterministe ou stochastique.
L'alignement fourni par ces méthodes est celui qui optimise une fonction objectif (F.O) choisie ou fonction score. Formulée mathématiquement, cette fonction permet une évaluation quantitative de la signification biologique et évolutionnaire d'un alignement. Sa valeur doit indiquer la relation entre les séquences du point de vue structure et évolution. Le choix d'une F.O est une tâche très délicate. Ceci est justifié par la difficulté même de définir une F.O qui capture fidèlement toute l'information biologique exhibée par un alignement donné. En d'autres termes, comment s'assurer mathématiquement qu'un alignement est correct biologiquement ? Ceci explique l'apparition d'un nombre non négligeable de F.Os telles que *SP* (Sum of Pairs) [12], *WSP* (Weighted Sum of Pairs) [4,13], *Coffee* [14], *Consensus* [15], *Entropie* [16] et *Log Expectation* [9]. Ces fonctions ont des caractéristiques différentes et permettent d'évaluer les alignements selon des aspects différents. L'optimum mathématique ne coïncidant pas souvent avec l'optimum biologique [1], ces fonctions permettent, à des degrés variés, de s'approcher de l'optimum biologique.
Reposant sur une seule F.O, les méthodes proposées dans la littérature fournissent des évaluations qui ne sont pas toujours concluantes. D'où l'idée d'utiliser plusieurs F.Os simultanément. L'optimisation multi objectif apparaît comme un cadre naturel pour mener cette étude. Elle permet non seulement la prise en compte de plusieurs F.Os simultanément mais aussi d'aboutir à un meilleur compromis se traduisant par une meilleure qualité des solutions et donnant lieu non pas à une seule solution potentielle mais plusieurs solutions potentielles. Ceci a l'avantage de fournir aux biologistes plus de choix au stade de prise de décision. En exploitant cette nouvelle idée, nous proposons dans ce papier une approche basée sur un algorithme génétique multi objectif. Pour cela, nous définissons un schéma de représentation pour coder les alignements, des opérateurs appropriés pour faire évoluer une population initiale et

une stratégie de sélection permettant de maintenir les bonnes solutions obtenues au cours de l'évolution.

Dans un souci de clarté, le reste du papier est organisé comme suit : La section 2 fournit une description de l'alignement multiple de séquences. La section 3 présente les concepts de base de l'optimisation multi objectif. Nous y trouverons une brève présentation des algorithmes évolutionnaires multi objectif. La section 4 est dédiée à la description de l'approche proposée. Les résultats expérimentaux feront l'objet de la section 5. Le papier s'achèvera par une conclusion et des perspectives.

2. Alignement Multiple de Séquences

L'analyse des séquences est un des thèmes principaux en bioinformatique où l'alignement de séquences représente une tâche de base. Une séquence quelle soit nucléique (ADN) ou protéique (protéine) est un agencement d'éléments de base (dits résidus) tels que les nucléotides (Adénine, Cytosine, Guanine, Thymine) dans le cas des séquences d'ADN (voir fig.1). L'alignement vise à comparer des séquences. Il est dit multiple s'il concerne plus de deux séquences. La comparaison vise à établir des similarités/divergences ou des homologies. Deux séquences sont identiques si elles possèdent une ressemblance parfaite. La similarité est une quantité qui se mesure en pourcentage d'identité. L'homologie quant à elle est une propriété de séquences qui a une connotation évolutive. Deux séquences sont dites homologues si elles possèdent un ancêtre commun. L'alignement multiple des séquences consiste à trouver le meilleur agencement des différents résidus colonne par colonne de telle sorte qu'il y ait le maximum de colonnes correctement alignées (voir fig.1). Afin d'aboutir à une meilleure mise en correspondance des séquences des vides ou des gaps sont insérés entre les résidus. Ce mécanisme est appelé « insertion/délétion ».

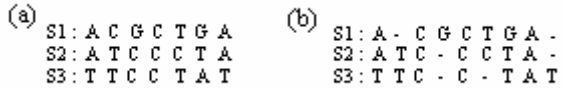


Figure 1 : Exemple d'alignement de séquences : (a) ensemble de trois séquences à aligner, (b) alignement obtenu par insertion de gaps.

3. Optimisation Multi Objectif

L'optimisation multi objectif (MOO) ou multicritère est un domaine dont l'importance est justifiée par la nature multi objectif des problèmes d'optimisation à résoudre dans la réalité. Elle se propose de traiter les problèmes qui nécessitent la satisfaction de plusieurs critères en même temps. Ces critères sont parfois conflictuels et parfois complémentaires. Un des aspects caractérisant l'optimisation multi objectif est qu'elle fournit un ensemble de solutions réalisant le meilleur compromis entre les

critères considérés. Cet ensemble de solution est appelé dans la littérature l'ensemble optimal de Pareto.
On considère un problème multi-objectif ayant *m* paramètres (variables de décisions) et *n* critères (objectifs à optimiser). Une optimisation multi-objectif revient à [17]:

Minimiser	$\vec{f}(\vec{x})$	(fonctions objectif à optimiser)
avec	$\vec{g}(\vec{x}) \leq 0$	(m contraintes d'inégalité)
et	$\vec{h}(\vec{x}) = 0$	(p contraintes d'égalité)

On $\vec{x} \in \Re^n$, $\vec{f}(\vec{x}) \in \Re^k$, $\vec{g}(\vec{x}) \in \Re^m$ *et* $\vec{h}(\vec{x}) \in \Re^p$. $\vec{x}$ est appelé vecteur de variables de décision. $\vec{f}$ regroupe les *k* fonctions objectif à optimiser.

3.1 Les Concepts Pareto

Nous donnons dans ce qui suit quelques définitions nécessaires à la compréhension de certains concepts de base que nous utiliserons dans la suite.

Définitions :

1. On dit qu'un vecteur de décision $\vec{x} \in \Re^n$ domine vecteur de décision $\vec{y} \in \Re^n$ si et seulement si :

$$\forall\, i \in \{1,\ldots,k\} : fi(\vec{x}) \leq fi(\vec{y}) \quad \wedge\ \exists\, j \in \{1,\ldots,k\} \text{ tel que } fj(\vec{x}) < fj(\vec{y}) \qquad (1)$$

2. *P* est dit ensemble optimal de Pareto si et seulement si :

   P est un sous ensemble de $\Re^n$: $P \subseteq \Re^n$

   $\forall\, \vec{x} \in P$, il n'y a pas un vecteur $\vec{y} \in \Re^n$ tel que $\vec{x}$ *est* dominé par $\vec{y}$

L'ensemble correspondant à l'ensemble optimal de Pareto dans l'espace des objectifs est appelé l'ensemble des solutions non dominées ou le *Front Pareto* (ou encore la frontière de Pareto).

3.2 Algorithmes Évolutionnaires Multi Objectif

Plusieurs méthodes ont été développées pour résoudre des problèmes multi objectif. Elles se basent sur l'utilisation de métaheuristiques telles que la recherche tabou, le recuit simulé ou les algorithmes évolutionnaires. Les algorithmes évolutionnaires multi objectif (MOEA : Multi Objectif Evolutionary Algorithm) offrent par le biais de la notion de population et les opérateurs génétiques (Croisement et Mutation), des mécanismes pertinents pour approcher en une seule exécution le front Pareto optimal. Les MOEAs sont très bien adaptés au traitement d'un problème d'optimisation multi objectif. Ce domaine est en évolution constante et ne cesse de se développer. Il existe plusieurs versions des MOEAs. Chacune tente d'apporter plus d'efficacité et de trouver des solutions les plus proches de l'optimalité. Lors de l'utilisation d'un MOEA deux problèmes essentiels sont à résoudre [18]:

- Comment calculer la valeur de fitness d'un individu et quelle est la procédure de sélection qu'il faut utiliser afin de s'approcher au maximum de la frontière de Pareto.
- Comment maintenir la diversité au sein de la population, empêcher une convergence prématurée et par conséquent atteindre une frontière de Pareto distribuée et uniforme.

Dans les MOEAs on trouve d'un côté, les méthodes agrégatives qui essayent de combiner les différentes F.Os en une seule équivalente. Le problème de MOO sera ramené à un problème d'optimisation mono objectif. De l'autre côté, les méthodes non agrégatives, essayent de trouver les solutions optimales en optimisant chaque F.O séparément. Par ailleurs, on peut distinguer les MOEAs élitistes et non élitistes. En raison de leur aspect stochastique, les MOEAs peuvent ne pas conserver les bonnes solutions rencontrées pendant leur exécution et les rendre disponibles à la fin de l'exécution de l'algorithme. L'élitisme assure simplement que les meilleurs individus de la population actuelle seront pris dans la prochaine génération. Il a été prouvé que la stratégie d'élitisme augmente de manière significative les performances des MOEAs [19]. L'élitisme peut être traduit par le fait de stocker les meilleures solutions dans une population secondaire qui servira comme une réserve pour produire de nouvelles solutions pour les générations futures.
Une autre différence peut être observée au niveau des différents MOEAs. Elle se situe dans la technique de sélection utilisée. Une des techniques la plus utilisée est le *ranking* où les solutions sont classées selon la dominance de Pareto. Celles qui ont le meilleur rang sont alors considérées optimales. Les dernières améliorations et suggestions apportées aux MOEAs favorisent l'exploration des techniques de *Sharing* ou heuristiques de partage [20] et qui sont souvent exprimées sous différents aspects tels que le *Niching*, *Clustering* [21] ou *Crowding* [19]. Ces techniques permettent le maintient de la diversité au sein de la population. Elles augmentent souvent la complexité des algorithmes mais garantissent en parallèle une meilleure configuration de la frontière de Pareto.

Nous nous intéressons ici aux MOEAs non agrégatifs et élitistes dont la sélection est basée sur la dominance de Pareto. Plusieurs expériences ont montré qu'elles sont les plus appropriées afin de bien approcher (construire) un front Pareto uniforme et bien distribué. Dans cette catégorie on peut citer PAES [22], Memetic-PAES [23], PESA2 [24], SPEA2 [25], et NSGA2 [19]. Les algorithmes génétiques multi objectif (MOGAs :MultiObjective Genetic Algorithms) sont une classe des MOEAs. Ils ont la particularité d'utiliser un codage binaire et mettre en œuvre les opérateurs génétiques traditionnels : le croisement et la mutation.

4. Un Algorithme Génétique Multi Objectif pour le MSA

L'idée nouvelle, sous jacente à ce travail, est de traiter le problème de MSA comme un problème d'optimisation multi objectif où différentes fonctions de score sont optimisées simultanément. L'utilisation de plusieurs F.Os nous permet de capturer les meilleures caractéristiques de chaque F.O. Le travail consiste précisément à définir

un algorithme génétique multi-objectif élitiste non agrégatif permettant d'optimiser plusieurs F.Os et de converger vers un ensemble de Pareto uniforme et distribué.

4.1 Formulation d'un MSA Multi Objectif

Un MSA Multi Objectif (MSAMO) peut être formulé de la manière suivante :

- Soit Π un ensemble d'alignements potentiels.
- Soient $f_1, f_2, \ldots, f_n$ un ensemble de n objectifs ou mesures de score.
- Il s'agit de trouver l'alignement optimal $A \in \Pi$ tel que :

$$A \text{ optimise } \{f_1(A), f_2(A), \ldots f_n(A)\}$$

Dans ce qui suit, trois mesures de score sont utilisées comme fonctions objectif : la mesure *WSP,* la mesure *Entropie* et la mesure *Consensus*. Il s'agit alors de résoudre le problème suivant:

$$\begin{cases} Maximiser\ WSP(A) \\ Maximiser\ (-Entropie(A)) \\ Maximiser\ Consensus(A) \end{cases}$$

4.2 Description de l'Approche

Notre approche est un algorithme génétique multi objectif qui admet un ensemble d'alignements qu'il fait évoluer via des opérateurs génétiques (croisement, mutation et sélection) pour aboutir à un ensemble d'alignements de bonne qualité voir « optimaux » après un certain nombre de générations. Notre algorithme maintient trois types de populations : la population initiale définie au début, la population courante définie à chaque génération et la population secondaire utilisée comme archive pour garder trace des meilleurs solutions trouvées au cours des itérations. Elle sert comme source de production pour chaque population courante.

4.3 Représentation d'un Alignement

Un individu étant un alignement potentiel. On doit donc en premier lieu trouver une représentation appropriée permettant la mise en œuvre de l'algorithme multi objectif. Le schéma de représentation que nous avons adopté est le codage binaire, où l'on associe la valeur 1 à un résidu et 0 au gap. Ainsi le codage d'un individu se traduit par une matrice binaire dont les dimensions dépendent du nombre de séquences à aligner et la longueur maximale obtenue après insertion d'un certain nombre de gaps. La figure (fig.2) suivante illustre le codage adopté :

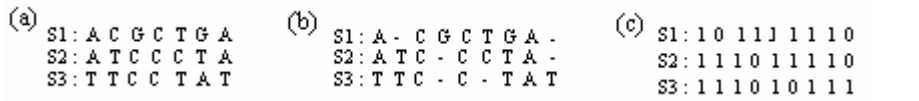


Figure 2 : Codage des alignements. (a) séquence initiale, (b) alignement potentiel, (c) codage binaire correspondant.

4.4 Les Fonctions Objectif Utilisées

Dans le cadre de notre travail, trois fonctions objectif ont été envisagées :

4.4.1 WSP (Weighted Sum of Pairs)

Cette mesure de score [13] consiste à attribuer des poids aux différentes paires de séquences à aligner. Ces poids peuvent être obtenus en dressant un arbre phylogénétique reliant ces séquences selon leurs similitudes ou distances. Cette fonction objectif est définie comme suit :

$$WSP(\mathrm{A}) = \sum_{i=1}^{K-1} \sum_{j=i+1}^{K} \sum_{h=1}^{L} Wij * sc(A[i,h], A[j,h]) \quad (2)$$

Avec *sc(A[i,h], A[j,h])* = le score de la paire de résidus alignés dont les coordonnés sont (i,h) et (j,h). *Wij* est le poids des séquences i et j, déterminé à partir de l'arbre phylogénique reliant les K séquences.

4.4.2 La Fonction Consensus

Il est parfois préférable de passer par une séquence consensus *C* pour évaluer la qualité d'un alignement multiple de plusieurs séquences [15]. L'idée consiste donc à trouver une séquence *C* : C1 C2….CL où L est la longueur de l'alignement, de telle sorte que chaque résidu c_i de *C* minimise le score de l'alignement.

Définition : Ayant un alignement de N séquences S1' S2' S3'….. SN', le caractère consensus c_i d'une colonne i (i = 1,…L) est celui qui minimise la somme des distances entre c_i et les résidus de la même colonne dans toutes les séquences.

$$d(i) = \sum_{j=1}^{N} d(Sj'[i], c_i) \quad (3)$$

La séquence consensus étant C : c_1c_2 c_3 …c_L, les erreurs d'alignement sont définies par $\sum_{i=1}^{L} d(i)$. Dans le cas d'une fonction de similitude, on cherche une séquence C, qui maximise le score.

4.4.3 La Mesure Entropie

La mesure d'entropie en MSA est la somme des entropies des colonnes [16]. L'entropie, introduite par *Shannon,* est en générale une mesure de variation des informations utilisée souvent dans la théorie de l'information. Pour chaque colonne d'un alignement multiple, l'entropie est calculée par la formule suivante :

$$\text{Entropie}(\mathrm{A}[:][i]) = -\sum_{a} c_{ia} * \log(p_{ia}) \quad (4)$$

Où c_{ia} est le nombre d'apparition du symbole *a* dans la colonne A[:][i], p_{ia} est la probabilité du symbole *a* dans la colonne *i :*

$$p_{ia} = c_{ia} / \sum_{a'} c_{ia'} \quad (5)$$

Plus la colonne est variable, plus l'entropie est grande. Le but est donc de trouver l'alignement qui minimise l'entropie de l'ensemble des colonnes d'un alignement.

4.5 La Population Secondaire

Pour garder trace des bonnes solutions obtenues au cours des générations, on utilise une population secondaire (PS) composée essentiellement des solutions optimales au sens de Pareto. Ce mécanisme garantit la survie des meilleurs individus et les protège contre toute élimination non souhaitée. Dans PS, on dépose périodiquement de nouvelles solutions non dominées (voir définitions section 3.1) générées par la population courante et on retire les solutions qui sont devenues dominées après insertion. Une réévaluation des individus de PS est obligatoire. Si après insertion, la taille de PS dépasse une taille maximale fixée alors une réduction devient nécessaire. Les individus de PS participent à la production des nouveaux individus pour les générations futures.

4.6 Évaluation des Individus

Afin de pouvoir sélectionner les meilleurs individus de la population courante puis enrichir PS, il faut tout d'abord évaluer ces individus afin d'établir un ordre d'efficacité. A chaque individu, est associé un vecteur de taille égale au nombre des F.Os utilisées et contenant les valeurs correspondantes. L'ordre d'efficacité est établi comme suit : l'individu est non dominé au sens Pareto d'après l'équation (1), son rang est alors égal à 'un'. Si par contre, un individu est dominé par un autre individu et par un seul, son rang est 'deux' et ainsi de suite jusqu'au dernier. Le rang d'un individu dépend du nombre d'individus qui le dominent. L'algorithme ci dessous illustre le mécanisme d'établissement des rangs.

L'algorithme MSArang
A : Solution à comparer (vecteur de dimension n = nombre des F.Os)
SC: Ensemble des solutions
$A_{rang} = 1$;
Pour tout $B \in SC$ *et* $B \neq A$
Si $B_1 > A_1$ *et* $B_2 > A_2 \ldots\ldots\ldots$*et* $B_n > A_n$ *alors* A_{rang} ++ ;
Fin si / sinon les solutions sont incomparables*
Fin pour
Fin MSArang

4.7 La Sélection

Cette opération permet de sélectionner les individus aptes à la reproduction et donc participer aux opérations de croisement et de mutation. Les individus aptes sont souvent les mieux classés lors de l'évaluation. Dans le cadre de notre approche, nous

avons utilisé la technique de sélection basée sur le rang (ranking) par l'exploitation de l'algorithme *MSArang* afin d'établir un rang entre les individus de la population, puis sélectionner ceux qui ont le rang égal à '1'. Ces derniers composeront le front Pareto.

4.8 Les Opérateurs Génétiques: Croisement & Mutation

Le croisement est une opération classique des algorithmes génétiques. Elle consiste en la combinaison du matériel génétique de deux individus de la population et en produire deux nouveaux individus (Enfants) héritant des caractéristiques de leurs parents. La position du croisement sur les deux individus est déterminée aléatoirement. Il est important de s'assurer que les alignements produits ne sont pas une copie des alignements initiaux ni identiques à d'autres alignements déjà insérés dans la population.

La mutation consiste à permuter un gap et un résidu ou un ensemble de gaps consécutifs avec un autre ensemble de résidus de même taille et voisins dans la même séquence ou sur plusieurs séquences. La mutation introduit plus de diversité dans les alignements nouvellement produits et parfois elle est exploitée seule pour générer de nouveaux individus lorsque le croisement n'est pas possible. Plusieurs opérations de mutation ont été exploitées [4] afin de s'assurer que chaque opération de mutation fournit réellement un nouvel individu.

4.9 Réduction de la Population Secondaire

Pour des raisons des coûts en temps et en espace, la taille de la population secondaire (*PS*) doit être fixe. Lorsque le nombre des solutions non dominées dépasse la taille permise, il faut sélectionner parmi les individus non dominés ceux qui vont être éliminés. Cette réduction ne doit pas entraîner ni une perte des bonnes solutions ni une perte de diversification. Il faut alors choisir soigneusement les éléments à éliminer à fin de conserver un front Pareto uniforme et distribué. La technique utilisée pour assurer une bonne réduction au niveau de notre *PS* est le *Clustering* [22] basée sur la distance de *Hamming*. L'algorithme a une complexité d'ordre $O\ (N \log N)$ avec N est la taille de *PS*.

4.10 L'Algorithme Génétique Multi objectif pour l'Alignement Multiple

Ayant défini les éléments de base de l'approche proposée, l'algorithme qui suit résume son fonctionnement général. Il a une complexité de $O(N(4MM'+ M'LogM'))$ avec N le nombre de générations ou itérations, M la taille de la population courante et M' la taille de la population secondaire.

Pseudo code de l'algorithme proposé
Début
-t=0 ;
-Initialiser la population P0 ;
-Évaluer individus de la population P0 selon les trois objectifs;
-Affecter un rang aux individus de P0 selon la dominance de Pareto ;
-Créer la population secondaire P' à partir des individus non dominées de P0 ;
- Si taille de P' dépasse un seuil alors Réduction de PS ;
Répéter
t = t+1 ;
Créer la population de Pt à partir de P' par mutation et croisement ;
Évaluer individus de la population Pt selon les trois objectifs;
Affecter un rang aux individus de Pt selon la dominance de Pareto ;
Sélection dans Pt en fonction de leur rang ;
Mise à jour de P' à partir des individus non dominés de Pt ;
Si taille de P' dépasse un seuil alors Réduction de PS ;
Jusqu'à nombre maximum d'itérations
Fin.

5. Résultats Expérimentaux

Pour évaluer les performances de la méthode proposée, plusieurs datasets de *Balibase* ont été utilisés. Balibase est une base des données biologiques de références construite par J. Thompson et al. [26] à des fins de validation. Elle fournit un programme d'évaluation Bali_score calculant les scores CS (% colonnes correctement alignées) et SPS (% des paires de résidus correctement alignées). D'autre part et pour générer les solutions initiales, nous avons exploité les méthodes : T_Coffee, Muscle, Align_m et ClustalX. La taille de la population initiale est fixée à 20 individus et celle de la population secondaire est 6 (≈ 1/3 de la population initiale). Les tables 1 et 2 présentent les résultats obtenus après 500 générations. On remarque qu'il y a une amélioration au niveau de toutes les F.Os et les scores CS et SPS.

Dataset=1hfh_ref1	T-coffee	Muscle	Align_m	Clustal X
Rang	2	**1**	4	3
WSP * 10^3	3.589	3.627	2.016	3.138
Entropie	135.94	132.71	189.97	137.18
Consensus	1795	1811	1009	1583
CS (Balibase)	0.839	**0.887**	0	0.758
SPS (Balibase)	0.902	**0.932**	0.270	0.886

Table1

Sol1	Sol 2	Sol 3	Sol 4	Sol 5
1	1	1	1	1
3.657	3.668	**3.699**	**3.699**	3.709
132.3	132.4	**132.6**	**132.9**	133.0
1828	1838	**1847**	**1847**	1852
0.903	0.903	**0.903**	**0.903**	0.903
0.941	0.941	**0.941**	**0.941**	0.941

Table2

Les alignements 3 et 4 (sol 3 et sol 4) ont eu, en général, les mêmes scores. Sont-ils des clones d'une même solution ? Non, car avec un peu d'observation, on remarque qu'il y a une différence du point de vue génotype qui peut être significative point de vue biologique (fig.3).

Alignement (3)

NETTC---YMGK **W S S - -** PPQC--EGLPC
NSGSISTCLRNG **W S A - -** QPICINSESKC
EKIINC—SLSGK **W S V - A** PPTC--EEARC
PSTTCLVSGNNT **W D K K V** APIC— E I ISC
NSGVLC---SGGE **W S D - -** PPTC--QIVKC

Alignement (4)

NETTC---YMGK **W - S S -** PPQC--EGLPC
NSGSISTCLRNG **W - S A -** QPICINSESKC
EKIINC—SLSGK **W – S V A** PPTC--EEARC
PSTTCLVSGNNT **WD K K V** APIC---E I ISC
NSGVLC---SGGE **W - S D -** PPTC--QIVKC

Figure 3

La figure 4 montre le graphe du front Pareto obtenu de l'exemple cité. Dans ce graphe, sont présentées uniquement les valeurs des fonctions WSP et Entropie. On peut y observer une frontière de Pareto uniforme et distribuée.

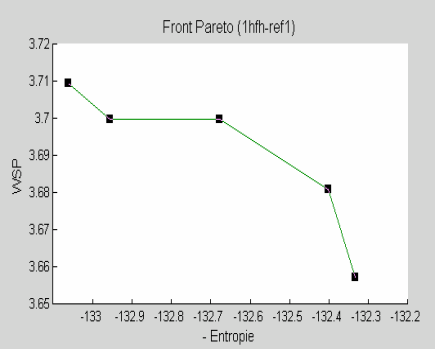


Figure 4 : Front de Pareto obtenu

6. Conclusion

Nous avons présenté dans cet article, une approche évolutionnaire multi objectif élitiste non agrégative permettant d'évaluer et de sélectionner les meilleures solutions possibles puis les raffiner par application d'opérateurs génétiques. La sélection utilisée est basée sur la notion de dominance de Pareto. Les solutions sont évaluées selon trois objectifs : WSP, Entropie et Consensus. L'approche développée a été testée sur plusieurs datasets de Balibase. Les résultats obtenus sont satisfaisants. Il serait intéressant d'étudier d'autres fonctions objectif dans un contexte multi objectif pour identifier l'ensemble le mieux adapté pour l'alignement des séquences.

Références

[1]. C. Notredame, "Recent progress in multiple sequence alignment: a survey", Pharmacogenomics, Vol. 3 no. 1, 2002.

[2]. M. Wallace, O.O'Sullivan, D.G. Higgins and C.Notredame, "M-Coffee: combining multiple sequence alignment methods with T-Coffee", Nucleic Acids Res., Vol. 34, No. 6, pp.1692–1699, 2006.

[3]. J.D Thompson, D.G. Higgins and T.J. Gibson, "CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice" Nucleic Acids Res. Vol. 22 no. 22 pp. 4673-4680, 1994.

[4]. C. Notredame and D.G. Higgins," SAGA: Sequence alignment by genetic algorithm", Nucleic Acids Res. Vol. 24 no. 8 pp. 1515-1524, 1996.
[5]. O.Gotoh, "Significant Improvement in Accuracy of Multiple Protein Sequence Alignments by Iterative refinement as Assessed by Reference to Structural Alignments", J. Mol. Biol. 264. pp. 823-838, 1996.
[6]. B.Morgenstern, K.Frech, A. Dress and T.Werner, "DIALIGN: Finding local similarities by multiple sequence alignment", Bioinformatics, Vol. 14 no. 3 pp. 290-294, 1998.
[7] C.Notredame, D.G. Higgins and J. Heringa, «T-Coffee: A Novel Method for Fast and Accurate Multiple Sequence Alignment », J. Mol. Biol. 302, pp. 205- 217, 2000.
[8]. K.Katoh, K.Misawa, K.Kuma and T.Miyata, "MAFFT : a novel method for rapid multiple sequence alignment based on fast Fourier transform", Nucleic Acids Res. Vol. 30 no. 14 pp. 3059-3066, 2002.
[9]. R.C Edgar, "MUSCLE: multiple sequence alignment with high accuracy and high throughput", Nucleic Acids Res. Vol. 32 no. 5 pp. 1792-1797, 2004.
[10]. V. Walle, I.Lasters, and L. Wyns, "Align-m: a. new algorithm for multiple alignment of highly divergent sequences", Bioinformatics, 2004.
[11]. C.B. Do, M. Mahabhshyam, M. Brudno, and S. Batzoglou, "ProbCons: Probabilistic consistency-based multiple sequence alignment", Genome res. 15. 330-340. 2005.
[12]. H. Carrillo and DJ. Lipman, « The multiple sequence alignment problem in biology », SIAM J. Appl. Math. 48, pp. 1073-1082, 1988.
[13]. S.F.Altshul, R. J. Carroll and D.J. Lipman, "weights for data related by a tree", J. Mol. Biol. 207, 647-653, 1989
[14]. C. Notredame, L. Holm and D.G. Higgins, «Coffee: an objective function for multiple sequence alignments », Bioinformatics, Vol. 14 no. 5 pp. 407-422, 1998.
[15]. K.Reinert, "Introduction to multiple Sequence Alignment ", Algorithmische Bioinformatik, WS, 03, 10, November 2003.
[16]. H.B.Nicholas, A.J. Ropelewski and D.W. Deerfield, « Strategies for multiple sequence alignment », Biotechniques, Vol. 32, no. 3 pp. 572-591, 2002.
[17]. Y. Collete et P. Siarry, "Optimisation multiobjectif ", Eyrolles. pp. 20-22, 2002.
[18]. E. Zitzler, K.Deb and L.Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results", Evol. Comput. Vol. 8, no. 2, pp. 173-195, 2000.
[19]. K. Deb, A. Pratap, S. Agarwal and T.Meyarivan, «A fast and Elitist Multiobjective Genetic Algorithm: NSGA-II", IEEE Trans. Evol. Comput. Vol. 6, no. 2, pp. 182-197, 2002
[20]. D. Goldberg and J. Richardson, « Genetic algorithm with sharing for multimodal function optimisation », Proceedings of the second ICGA, pp. 41-49, 1987.
[21] E.Zitzler and L.Thiele," An Evolutionary Algorithm for Multiobjective Optimization: The Strength Pareto Approach", Technical Report 43, Computer Engineering and Communication Networks Lab (TIK), Swiss Federal Institute of Technology (ETH), Zurich, Switzerland, 1998.
[22]. J. D. Knowles and D. W. Corne. "The Pareto Archived Evolution Strategy: A New Baseline Algorithm for Multiobjective Optimisation.", IEEE, *Congress on Evolut. Comput.*, pp 98–105, 1999
[23]. J. D .Knowles and D. W. Corne, «M-PAES: a Memetic algorithm for multiobjective optimisation", In proceedings of the 2000 congress on Evol. Comput. pp. 325-332, 2000.
[24]. D.Corne and al, « PESA 2: Region-based selection in Evolutionary multiobjective optimisation", in proceedings of the genetic and evolutionary computation, San Francisco, California, pp. 283-299, 2001.
[25]. E.Zitzler and M.Laumanns, « SPEA 2: Improving the strength Pareto evolutionary algorithm", technical report, 103, 2001.
[26]. J.Thompson, P.Koehl, R.Ripp and O.Poch, "BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark", Proteins. Vol. 61, pp.127-136. 2005.

Beetweeness centrality of metabolic network using Elementary Net Systems

Hafida Raaf[1], Saïd Abdeddaïm[2], Belhadri Messabih[3]

[1,3]Département d'informatique faculté des sciences université des sciences et de la technologie Mohamed Boudiaf Oran Algérie

[1]e-mail HRaaf@univ-usto.dz Hafida_raaf@yahoo.fr

[2]Laboratoire d'informatique, de traitement de l'information et des systèmes EA 4051 INSA Rouen France

Abstract. Pathway analysis of large metabolic networks meets with the problem of combinatorial explosion of pathways we propose to deconstruct metabolic networks into sub-networks based on the global network structure. To do this we present a metabolic network with ENS (Elementary Net Systems). ENS are derived from low Petri nets "Conditions/events systems"(substances/enzymatic reactions). They are used to describe and analyse real-time systems as a set of cooperating systems. We present in this paper an approach to compute the beetweeness centrality index which is found in many networks. This measure can be used not only to detect community structure or sub-network structure but also to predict the most influential event in ENS. So we firstly give a new strategy to represent ENS network as an event depending graph. Secondly we give a new definition of shortest path to compute centrality indices on this depending graph**.**

Key words: Petri net, Elementary net systems, Shortest path, beetweeness, Biochemical network, Metabolic pathway.

1 Introduction

1.1 Generalities

The behaviour of Metabolic network can be modelled and analyzed by Petri Nets [12][11]. The introduction of the Petri nets to represent the biochemical reactions was carried out in 1993 by [12] and then others works followed [6], [5], [9], [10] they showed that the graphic structure (static) of the Petri nets easily (qualitatively) models the interactions in a cell: the places represent the substances, the transitions represent the reactions, the arrows the tokens and the weights of the arrows model the quantity of the substances which is subjected to a reaction to produce a quantity of products ensuring the expression of temporal evolution of these networks (dynamics). The execution of Petri nets allow the quantitative analysis (kinetic) [9] of these networks. Structural analysis of large metabolic network [11][17] meets with the problem of combinatorial explosion. We propose to deconstruct metabolic network into sub-networks based on the global network structure. Identifying the modular organization of metabolic network by certain network decomposition methods can help us better understand the organization principle of complex metabolic network. To do this we use ENS (Elementary Net System) to model a metabolic network.

Search on Petri nets among has focused on a number of distinctive structural properties that systems share. To Tackle the modelling problem raised by these system we consider a low level model . This model is called Elementary Net System (ENS) known as "Conditions/events systems" and have been defined by Thiagarajan [15] et Rozenberg[14][1] to describe easily intrinsic characteristics of real-time systems such as high degree of parallelism and synchronisation.
Although Petri nets and ENS where been defined several years ago, nothing was known about their decomposition into sub-networks or the must influential event. We propose to use the notion of beetweeness centrality used for detecting communities in social large network [4] to detect the most influential event in ENS. To find community (sub-network) structures in graphs, the interest in the domain has started with a new *divisive* approach proposed by Girvan and Newman [4], the edges with the largest *beetweeness* (number of shortest paths passing through an edge) are removed one by one in order to split hierarchically the graph into communities.

1.2 Centrality in ENS

An essential tool for the analysis of social networks are centrality indices defined on the vertices of the graph [3][16]. They are designed to rank the actors according to their position in the network and interpreted as the prominence of actors embedded in a social structure. Many centrality indices are based on shortest paths linking pairs of actors [4][2], measuring, e.g., the average distance from other actors, or the ratio of shortest paths an actor lies on. Many network-analytic studies rely at least in part on an evaluation of these indices. All of them are based on the same idea. If two communities are joined by only a few inter-community edges, then all paths through the network from vertices in one community to vertices in the other must pass along one of those few edges. Given a suitable set of paths, one can count how many go along each edge in the graph, and this number we then expect to be largest for the inter-community edges, thus providing a method for identifying them.
To apply this measure, we first represent ENS as a graph of events in which events are defined as nodes, and conditions as depending connections between the nodes. Secondly we give a new definition of shortest path to compute centrality indices on this graph**.** In the next section we recall basic definitions about ENS and we derive from the firing rule of event the general rule of depending events to define the events graph. As this graph is composed with two types of nodes, in section 3, we give a definition of shortest path. The fourth section is devoted to adapt the beetweeness algorithm built by [2] and finely we model metabolic network with ENS.

2 Elementary net systems

Elementary net systems are based on low level Petri nets Known as "conditions/events systems". Places also called conditions are allowed to contain at most one mark. Consequently, transitions, also called events, may fire not only if input places contain one mark but also if output places are empty. Obviously, such nets are bounded. Note that the firing takes no time, as for reactive systems.

Definition 1

An ENS is a tuple E= (G,M0) where G and M0 are the Petri part:

G=(S,R,Prec,Post) where

S is the set of conditions and R is the set of events , the set Prec of arrows represents directed link from pre-conditions to event and the set Post of arrows represents directed link from event to post-conditions.

M0 is the initial marking M: S→ {0,1} (number of marks in a condition of set S is zero or one)

-Prec(r) = °r where °r represent the set of preconditions needed by an event r, Prec ⊆ Rx party set(S).
-Post(r) = r° where r° represent the set of post-conditions of an event r , Post ⊆ Rx party set(S).

Example 1

Fig1 shows an elementary system with R1, R2, R3, R4, R5 are events presented by rectangles. s1, s2, s3, s4, s5, s6 are conditions presented by circles. And one mark in s1

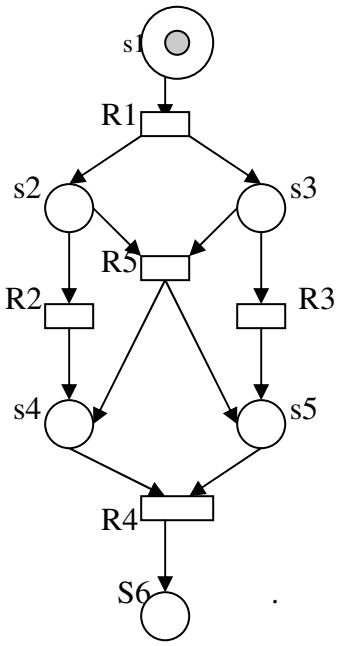


Fig1 : AN Elementary Net System

Definition 2

An event r is enabled iff all its pre-conditions are enabled

$$\forall s \in {}^\circ r \quad M(s)=1$$

The firing of an enabled event r from marking M results in marking M' according to the following:

$\forall s \in {}^\circ r$ M'(s)=0, $\forall s \in r^\circ$ M'(s)=1, $\forall s \in :S - ({}^\circ r \cup r^\circ)$ M'(s)=M(s)

3 Events Graph

A **directed event graph** is defined as follow :

i)A **directed link** from event node A to event node B is added iff all pre-conditions of B composed with post-condition of the event A. We say an event B **can occur (to fire)** only after A (if B is connected only with A).

ii) An event B can occur only after the events A and C iff pre-conditions of B composed with post-conditions of A and post-conditions of C.

iii) An even B can occur only after A1, A2…and An iff pre-conditions of B are composed with post conditions of A1 and A2 …An.

Definition 3

Let G = (S, R, Prec, Post) be an ENS. Gr = (R $\cup$ T, P) is the directed event graph of G, if the following conditions hold for all z and r in R, and all t in T :

- $(r, z) \in P$ iff $r1^\circ = {}^\circ r2$.
- $(z_i,t) \in P$ and $(t,r) \in P$ iff $z_i^\circ \cap {}^\circ r \neq \phi$ and $z^* \cap {}^\circ r = {}^\circ r$. With $z^* = \underset{i=1,2\ldots j}{\cup} z_i^\circ$

In a reaction graph Gr = (R $\cup$ T, P), we denote by Pre and Succ the following sets:

-Pre(r) = $\{u \in R \cup T / (u,r) \in P\}$.
-Succ(r)= $\{u \in R \cup T / (r,u) \in P\}$.

Example 2

Fig 2 shows the events graph of (fig 1)

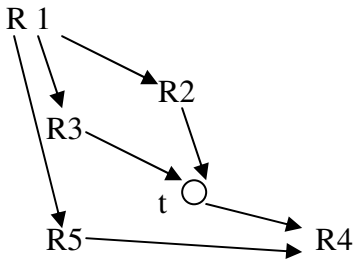


Fig2 : The event graph

4 Shortest path and beetweeness

4.1 Shortest path

A quantity of interest in many social network (modelled with a graph) studies is the ‘‘beetweeness’’ of an actor *i*, which is defined as the total number of shortest paths between pairs of actors that pass through *i*. This quantity is an indicator of who the most influential people in the network are, the ones who control the flow of information between most others. We are interesting to do this idea for detecting the most influential event in the event graph of an ENS.
As the event graph is composed with two types of nodes (R or T nodes) an adequate shortest path definition is necessary.

1) Let w be a weight function on the arrow.
We define

$w((u,r)) = 1 \qquad (u,r) \in P, u \in R \cup T, r \in R$ and
$w((r,t)) = 0 \qquad (r,t) \in P, r \in R, t \in T.$

Weights are used to measure, the strength of a link.
Define a path from $s \in R$ to $r \in R$ as an alternating sequence of vertices and edges, beginning with s (firing the event s) and ending with r (r is enabled) , such that each edge connects its preceding with its succeeding vertex. We use d(s,r) to denote the distance between vertices s and r, i.e. the minimum length of any path connecting s and r in Gr.
By definition, d(s,s) = 0 for every $s \in R$ and d(s, r) = d(r, s) for $s, r \in R$. We assume familiarity with standard algorithms for shortest-paths problems.

2) The minimal path of an event $t \in T$ to occur after firing an event s is the maximal path of a succeeding events such that pre-conditions of t are taken on and all events preceding t are firing and enabled with event s. We consider here the shortest-path as the longest succeeding path of events witch make some pre-conditions of t taken on. In other words

$d(s, t) = \underset{r}{\text{Max}}\, d(s, r)$; and for all event r preceding t $d(s,r) > 0$.

Definition 4

1) $d(s, s) = 0$ for every $s \in R \cup T$;
2) $d(s, t) = \underset{v}{\min} (d(s,v)) + 1$; with $(v,t) \in P$, $v \in R \cup T$, $s \in R \cup T$, $t \in R$.
3) $d(s, t) = \underset{r}{\text{Max}}\, d(s, r)$; with $(r,t) \in P$, $s \in R \cup T$, $r \in R$, $t \in T$;

4.2 Centrality Indices Based on Shortest Paths

Several measures capture variations on the notion of a vertex's importance in a graph. Let σ_{st} denote the number of shortest paths from $s \in R \cup T$ to $t \in R \cup T$, where $\sigma_{ss} = 1$ by convention. Let $\sigma_{st}(v)$ denote the number of shortest paths from s to t, that some $v \in R \cup T$ lies on.
Given pairwise distances and shortest paths counts, the pair-dependency

$$\delta_{st}(v) = \sigma_{st}(v) / \sigma_{st}$$

of a pair $s, t \in R \cup T$ on an intermediary $v \in R \cup T$, i.e. the ratio of shortest paths between s and t that v lies on, is given by:

$\delta st(v) = \quad 0 \quad$ if $d(s, t) < d(s, v) + d(v, t)$

$\delta st(v) = \quad \sigma_{sv} . \sigma_{vt}$ otherwise

To obtain the beetweeness centrality index of a vertex v, we simply have to sum the pair-dependencies of all pairs on that vertex,

$$CB(v) = \sum_{s \neq v \neq t \in R \cup T} \delta st(v)$$

Therefore, beetweeness centrality is determined [2] in two steps:

1. compute the length and number of shortest paths between all pairs
2. sum all pair-dependencies.

We modify the faster Algorithm defined by Brandes [2] according with our definition of shortest path. In Brand algorithm, the beetweeness centrality can be computed in O(nm + n2 log n) time and O(n +m) space for weighted graphs. Where n is the number of vertices and m the number of links. For unweighted graphs, running time reduces to O(nm). Brand count shortest paths using traversal algorithms. Both breadth-first search (BFS) for unweighted and Dijkstra's algorithm for weighted graphs start with a specified source vertice and, at each step, add a closest vertex the set of already discovered vertices in order to find shortest paths from the source to all other vertices. He determines the beetweeness centrality index by solving one single source shortest-paths problem for each vertex. At the end of each iteration, the dependencies of the source on each other vertex are added to the centrality score of that vertex. Our adaptation doesn't modify Brand's algorithm complexity. It takes in count tow types of nodes ($s \in R \cup T$) and applies definition 4 to determine the length of shortest paths between nodes as follow.

Algorithm 1: Beetweeness centrality in unweighted and directed graphs

```
CB[v] ← 0, v ∈ R ∪ T;
for s ∈ R ∪ T do
    S ← empty stack;
    P[w] ← empty list, w ∈ R ∪ T ;
    σ [t] ← 0, t ∈ R ∪ T; σ [s] ← 1;
    d[t] ←-1, t ∈ R ∪ T ; d[s] ← 0;
    Q ← empty queue;
     enqueue s → Q;
     while Q not empty do
            dequeue v ← Q;
            push v → S;
            foreach w ∈ Succ(v) do
             if w∈ R Then
                        // w found for the first time?
                        if d[w] < 0 then
                                    enqueue w → Q;
                                    d[w] ← d[v] + 1;
                        end
                        // shortest path to w via v?
                         if d[w] = d[v] + 1 then
                                                σ [w] ← σ [w] + σ [v];
                                                 append v → P[w];
```

```
                                    else
                                     if d[w]>d[v]+1 then
                                                    d[w]←d[v]+1;
                                                    σ [w]  ←  σ [v];
                                                    P[w] ←empty list
                                                    Append v → P[w]
                                    end
                        end
                       else // w in T //
                         if d[w]<0 then
                                    enqueue w → Q;
                                    d[w] ← d[v]
                         end
                         If d[w]<d[v] then
                                    d[w] ← d[v]
                                    σ [w]  ←  σ [v];
                                    P[w] ←empty list
                                    Append v → P[w]
                                   else
                                   if d[w] = d[v]  then
                                                σ [w]  ← σ [w] + σ [v];
                                                append v → P[w];
                                    end
                        end
              end
      end
δ[v]← 0, v ∈ R ∪ T;
// S returns vertices in order of non-Increasing distance from s
while S not empty do
pop w  ← S;
for v ∈ P[w] do δ [v]  ← δ [v] + σ [v]/σ [w] . (1 + δ [w]);
if w ≠ s then CB[w]  ← CB[w] + δ [w];
end
end
```

Example 3
Fig 3 shows the beetweeness index of (fig 2)

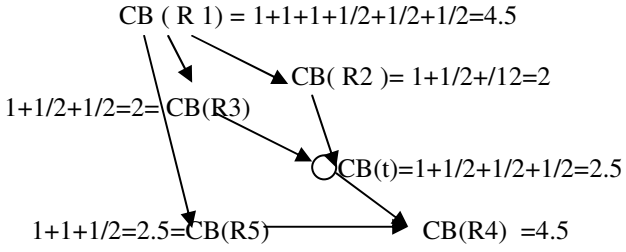


Fig3 : The beetweeness indices

5. Metabolic network

The metabolic networks are represented by the biochemists as complex graphs whose nodes constitute the metabolites and the arrows represent the reactions which are catalysed by specific molecules called enzymes. The structural(topological) analysis of **metabolic pathways** only uses information about stochiometric structure and the reversibility or irreversibility of enzymatic reactions. It consists in computing a minimal enzymes set that describe the conical steady state solution space for flux distribution in a metabolic network.
In the modelling of metabolic systems, usually a distinction is made between external and internal metabolites. External metabolites are the sources and sinks of the network; their concentrations are assumed to be buffered. Internal metabolites (intermediates) have to be balanced with respect to production and consumption at steady state. In many cases, there are specific biochemical reasons to treat a metabolite as internal (for example, when it is known to be merely an intermediate) or as external (e.g. nutrient or excreted product).
Classification of metabolites is not clear: an external metabolite is considered to be present in large excess so that its concentration is virtually unaffected by the reactions under study. An internal metabolite, however, must fulfil a balance equation implying that production is equals consumption. A first work to classify metabolites was proposed by Schuster *et al.*[17]. It considers the **metabolite degree of network connectivity.** Above a given connectivity threshold, some metabolites are considered as *external.* Then others works followed to decompose metabolic networks ([13][7] [8]). Identifying the modular organization of metabolic network by certain network decomposition methods can help us better understand the organization principle of complex systems. Here we represent a metabolic network as an ENS and than represent it as a **reaction depending graph**. The connectivity in this graph describes the sequences of enzymatic reactions and not only connectivity.

Definition 5

A metabolic network is represented as ENS=(S,R,Prec,Post) where S is the set of metabolites and R is the set of enzymatic reactions , the set Prec of arrows represents directed link from metabolite substrates to reactions and the set Post of arrows represents directed link from reactions to products.

-R=R_irev $\cup$ R_rev (irreversibles and reversibles enzymatics reactions)

-Prec(r) = °r where °r represent the set of substances needed by a reaction r, Prec $\subseteq$ Rx party set(S).
-Post(r) = r° where r° represent the set of products of a reaction r , Post $\subseteq$ Rx party set(S).

Example 4

the pentose-phosphate-pathway (PPP)

S={ G6P, NADP, 6PL, NADPH, H, H2O, 6GP, CO2, Ru5P, Ru5P, R5P, Xu5P, R5P,GAP, S7P, F6P, E4P}
R={ g6pdh, 6pgl, 6pgd, rpi, rpe, tkt, tal}
g6pdh: G6P + NADP → 6PL + NADPH + H
6pgl: 6PL + H2O →6GP + H
6pgd: 6GP + NADP → NADPH + CO2 + Ru5P
rpi: Ru5P ←→ R5P rpi is reversible
tkt: Xu5P + R5P ←→ GAP + S7P
tal: GAP + S7P ←→ F6P + E4P
tkt2: Xu5P + E4P←→ GAP + F6P

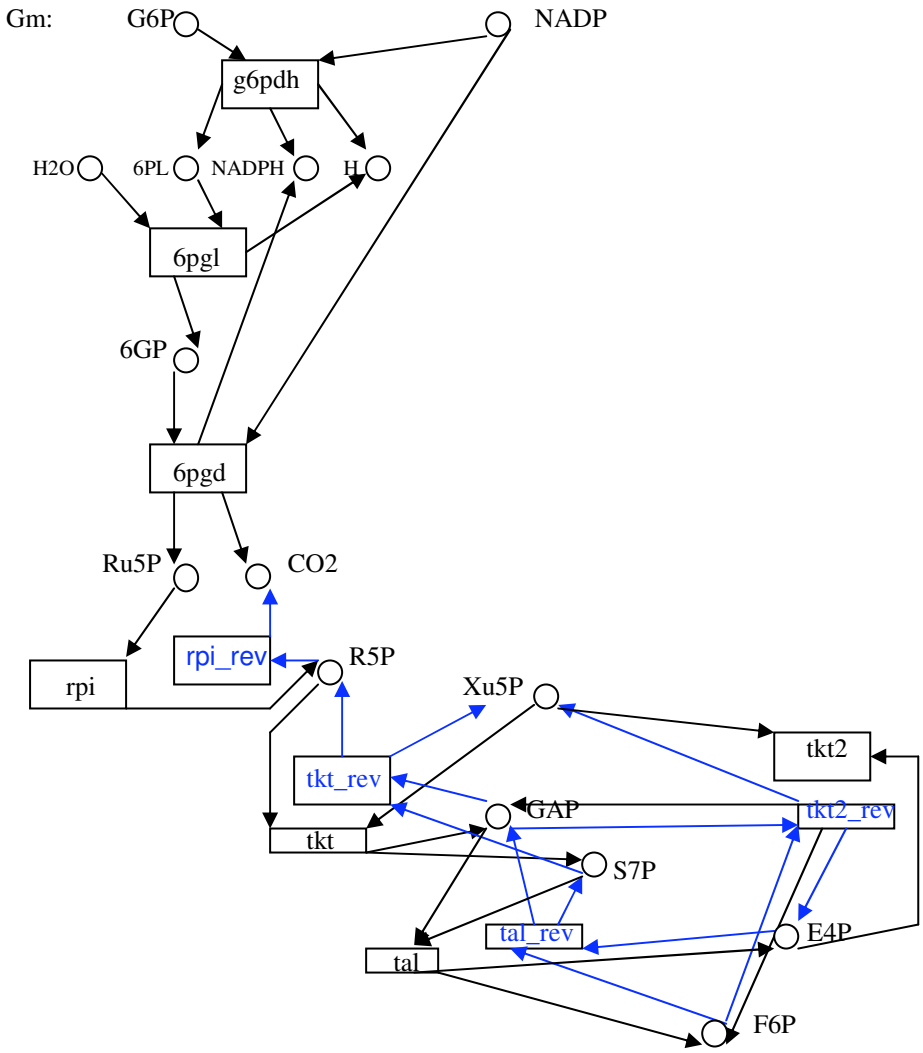


Fig 4: Metabolic network with ENS

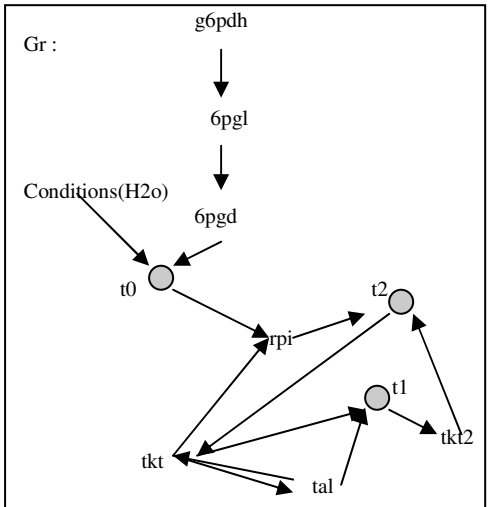


Fig 5 : The reaction depending graph

6. Conclusion

In this paper we have investigated a centrality structure in ENS. This beetweeness centrality is based on compute the length and number of shortest paths between all pairs events and sum all pair-dependencies. Our approach permit firstly to describe an ENS network as an event depending graph, secondly, to give a new definition of shortest path on graph witch defined with tow types of node and we have adapt the beetweeness algorithm for our definitions. The use of ENS to model metabolic network permit to represent the metabolic network as a reaction (event) depending graph.

7. References

1. C.Andre "synchronized Elementary Net Systems". Advances in Petri Nets 1989,Crzegorz Rozenberg editor, LNCS 424, Springer-Verlag.
2. Brandes, U., "A faster algorithm for betweenness centrality*", Journal of Mathematical Sociology,* vol.25, n° 2, 2001, p. 163-177.
3. Freeman, L. C. (1979). Centrality in social networks: Conceptual clarification. Social Networks, 1:215-239.
4. Girvan,M. and Newman,M.E.J. (2002) Community structure in social and biological networks. *Proc. Natl Acad. Sci. USA*, **99**, 7821–7826.
5. Goss, P. J. E. and Peccoud, J.(1998).*Quantitative modelling of stochastic system in molecular biology by using stochastic Petri nets*. Proc. Natl. Acad. Sci. USA. Vol.95,pp.6750-6755.
6. Hofestaedt, R. and Telen, S. (1998). Quantitative modelling of biochimical networks. In Silico. Biol. 1, 39-53.
7. Holme, P. , Huss,M. and Jeong,H. *Subnetwork hierar-chies of biochemical pathways*. Bioinformatics 19, 532– 538 (2003).
8. Hong-Wu Ma1, Xue-Ming Zhao, Ying-Jin Yuan and An-Ping Zeng1**,.** Decomposition of metabolic network into functional modules based on the global connectivity structure of reaction graph bioinf Vol. 20 no. 12 2004, pages 1870–1876
9. Matsuno, H. and Nagasaki et al. Hybrid Petri Net representation of Gene regulatory network. Pacific symp. On boiocom. 2000, 5 :338-349
10. Ming Chen : in silico systems analysis of biopathways. Thesis in silico biology.2004
11. Raaf ,H. Messabih, B Modelling and decomposition of metabolic network

In Proc. Conf. Inter sur l'informatique et ses applications Saïda 20006.
12. Reddy, V. N., Mavrovouniotis, M. L. and Liebman, M. N. (1993). Petri Net Representation in Metabolic Pathways. Proc. First Inter. Conf on Intel. Syst. For Molecular Biology, Hunter, L et al, AAAA Press, Menlo Park, pp,328-336
13. Ravasz,E., Somera,A.L., Mongru,D.A., Oltvai,Z.N. and Barabasi,A.L. (2002) Hierarchical Organization of modularity in metabolic networks. *Science*, **297**
14. G.Rozenberg: Behavior of Elementary Net Systems. In [ACPN86] Part I,60-94.
15. P.S.Thiagarajan: Elementary Net Systems. In [ACPN86] Part I,26-59.
16. M. E. J. Newman and M. Girvan, Finding and evaluating community structure in networks. Preprint cond-mat/0308217 (2003).
17. Schuster,S., Pfeiffer,T., Koch,I. Moldenhauer,F., and Dandekar,T. (2002) Exploring the pathway structure of metabolism: decomposition into subnetworks and application to *Mycoplasma pneumoniae*.*Bioinformatics*,18, 351– 361.

Performance analysis of feature interaction resolution methods

Ahmed Khoumsi

University of Sherbrooke, Dep. GEGI, Sherbrooke J1K2R1, CANADA
Ahmed.Khoumsi@USherbrooke.ca

Abstract. The designers of telecommunications services are confronted with the commonly called *Feature Interactions* (FI) problem: services (or features) that behave correctly alone, may behave incorrectly once they are used together. In this paper, we model and compare performances of two technologies for solving the FI problem, namely client-server (CS) and mobile agents (MA) technologies.
Keywords: Feature interaction resolution, performance modeling and analysis, client-server, mobile agent.

1 Introduction

The designers of telecommunications services are confronted with the following problem: services (or features) that behave correctly alone, may induce an undesired behavior once they are used together. This problem is commonly called *Feature Interactions* (FI) problem [1–7]. In the sequel, the term "FI problem" will be abbreviated by FI.

Solving a FI means finding a solution that prevents the corresponding undesired behavior. In this paper, we consider *distributed FIs*, that are the most difficult to solve because they involve services running in *different* components of the network. Their resolution necessitates communication between the components involved in the FI. Client-server (CS) and mobile agents (MA) are two examples of technologies for realizing the necessary communication. As we will see, the technology used has an influence in the performance in terms of cost and delay of the traffic generated in the network.

Note that the aim in this paper is *not* to present new methods for solving FIs, but it is to show how to *model, evaluate and compare performances of given MA-based and CS-based FI resolution methods*.

The rest of this paper is structured as follows. In Section 2, we present a simple example of distributed FI. Section 3 introduces two FI resolution methods, based on MA and CS technologies respectively, and illustrates them in the resolution of the FI of Sect. 2. In Section 4, we propose simple performance models for evaluating the *cost* of traffic and the *delay* of traffic in a network. Section 5 applies the models of Sect. 4 to evaluate and compare performances of the two resolution methods of Sect. 3 when they are applied to the FI of Sect. 2.

2 Example of distributed FI: OS-OCS

Let us present an illustrative example of distributed FI that will be used throughout this paper. Two services OS and OCS are involved, and the resulting FI is denoted OS-OCS.

Operator Service (OS): Every subscriber can use the OS service which acts like an outgoing "normal" call, except that it is operator-assisted.

Originating Call Screening (OCS): The OCS service allows to screen outgoing calls based on the destination number. More precisely, a OCS subscriber $\mathcal{A}$ can put numbers in a screening list L_{ocs}, and then the service OCS blocks any attempt of $\mathcal{A}$ to call a subscriber whose number is in L_{ocs}.

FI OS-OCS: When OS and OCS are used together, a FI OS-OCS occurs because the intention of a OCS subscriber may be not respected. In fact, let us assume that $\mathcal{A}$: 1) is subscriber to OCS, 2) has put in L_{ocs} the number of a subscriber $\mathcal{B}$, and 3) tries to call $\mathcal{B}$ by using OS.
The OS operator allows $\mathcal{A}$ to call $\mathcal{B}$, because she has not access to the content of L_{ocs}.

3 Use of MA and CS for solving FIs

3.1 Approach and Generic operations

Our approach of FI resolution is inspired from [8], where a set of generic operations is defined with the idea that FIs can be solved if the generic operations are available as a software library and allowed to be executed at will in each component hosting services that may be involved in FIs.

Let us first present the main generic operations of [8], where each operation is presented as a software procedure that can have input and output argument(s) which are prefixed by **in** and **out**, respectively. When necessary, certain procedures are presented more than once, for different values of their arguments. In the definition of a procedure executed in a given component $\mathcal{C}$: "incoming call" means "callee part (in $\mathcal{C}$) of a call process", "outgoing call" means "caller part (in $\mathcal{C}$) of a call process". A variable representing an argument is in lower case, and a specific value of an argument (i.e., a constant) is in upper case. Note that each function returns its result as an output argument. Here are the main generic procedures:

- *Interrupt(***in***:INC-CALL,***in***:point)* and *NoInterrupt(***in***:INC-CALL)*:
  After the execution of *Interrupt(***in***:INC-CALL,***in***:point)*, every incoming call will be interrupted at a given *point* of the call process.
  *NoInterrupt(***in***:INC-CALL)* cancels the effect of a previous *Interrupt(***in***:INC-CALL,***in***:point)*.
- *Interrupt(***in***:OUTG-CALL,***in***:point)* and *NoInterrupt(***in***:OUTG-CALL)*:
  After the execution of *Interrupt(***in***:OUTG-CALL,***in***:point)*, every outgoing call will be interrupted at a given *point* of the call process.
  *NoInterrupt(***in***:OUTG-CALL)* cancels the effect of a previous *Interrupt(***in***:OUTG-CALL,***in***:point)*.

- *Resume(***in***:point)* resumes the last interrupted (incoming or outgoing) call. The call is resumed from a given *point* which may be different from the point where the call has been interrupted.
- *Write(***in***:content,***in***:container)* replaces the content of a given *container* by a given *content*.
- *Read(***in***:container,***out***:content)* reads the *content* of a given *container*, and returns the *content*.
- *IsSubscTo(***in***:local-user,***in***:service,***out***:boolean)* returns a boolean value indicating whether a given *local-user* is subscriber to a given *service*.
- *GetInfo(***in***:local-user,***in***:service,***out***:information)* returns all the relevant *information* related to a given *service* of a given *local-user*.
- *SendMsg(***in***:local-user,***in***:message)* sends a given *message* to a given *local-user*.
- *IsInLoop(***in***:path,***out***:boolean)* returns a *boolean* value indicating whether *path* contains a loop.
- *Move(***in***:destination)* is used by a mobile agent to move to a given *destination* site.
- *Move(***in***:content,***in***:destination)* is used by a mobile agent to move to a given *destination* site carrying the information *content* with it.

The resolution of a distributed FI necessitates communication between the components involved in the FI. Mobile agents (MA) and Client-Server (CS) are two well known technologies for realizing the necessary communication. The authors of [8] study the use of MA and CS for solving FIs. Let us illustrate the use of these two technologies for the resolution of the FI OS-OCS introduced in Sect. 2. In the sequel, the term *call* will have two meanings: a *function call* (in programming language) or a *phone call*. For avoiding any ambiguity, the function call is written in capitals (i.e., CALL).

3.2 Resolution of OS-OCS

Services OS and OCS and a FI OS-OCS are presented in Section 2. This FI will be solved by using an approach consisting of *prioritizing* one of the two services. If priority must be given to OS, then actually there is no FI, because OS is already given priority in the behavior described in Section 2.

Let us now show how the FI can be solved if priority must be given to OCS. The FI is then solvable if the OS operator can have access to the content of the L_{ocs} of $\mathcal{A}$ (denoted $L^{\mathcal{A}}_{ocs}$). An approach is to insert the content of $L^{\mathcal{A}}_{ocs}$ into the L_{ocs} of the OS operator (denoted L^{OS}_{ocs}). For that purpose, we assume that the OS operator is subscriber to OCS. Let us present a MA-based method and a CS-method using this approach for solving OS-OCS. We define the function *GetInfoIfSubscToOCS(***in***:*$\mathcal{A}$*,***out***:y)* as follows:

CALL *IsSubscTo(***in***:*$\mathcal{A}$*,***in***:OCS,***out***:b)*
If b = True then : CALL *Read(***in***:*$L^{\mathcal{A}}_{ocs}$*,***out***:y)*
Else : $y = \emptyset$.

This function returns the variable y that contains the screening list of $L^{\mathcal{A}}_{ocs}$ if $\mathcal{A}$ is subscriber to OCS, and is empty otherwise.

Use of MA technology for solving OS-OCS:
A mobile agent (MA) in the component of OS makes a local CALL of *Interrupt(***in***:INC-CALL,***in***:BEFORE-CHECK-LOCS)* and since then, every incoming call will be interrupted at a point before L_{ocs}^{OS} is checked. As soon as an incoming call from a user $\mathcal{A}$ is interrupted in point *BEFORE-CHECK-LOCS*, the actions taken by the MA are:

1. Make a local CALL of *Move(***in***:*$\mathcal{A}$*)* to move to the component of the caller $\mathcal{A}$.
2. Make a local CALL of *GetInfoIfSubscToOCS(***in***:*$\mathcal{A}$*,***out***:y)* to obtain the screening list of $\mathcal{A}$ (in y) if $\mathcal{A}$ is subscriber to OCS.
3. Make a local CALL of *Move(***in***:y,***in***:OS)* to return to OS carrying the information y.
4. If $y = \emptyset$ (i.e., $\mathcal{A}$ is *not* subscriber to OCS): make a local CALL of *Resume(***in***:BEFORE-CHECK-LOCS)* to resume the previously interrupted incoming-call.
5. If $y \neq \emptyset$ (i.e., y contains $L_{ocs}^{\mathcal{A}}$): make a local CALL of:
   - *Write(***in***:y,***in***:*L_{ocs}^{OS}*)* to obtain L_{ocs}^{OS}=$L_{ocs}^{\mathcal{A}}$
   - *Resume(***in***:BEFORE-CHECK-LOCS)* to resume the previously interrupted incoming call.

Use of CS technology for solving OS-OCS:
Instead of using a MA that moves between $\mathcal{A}$ (subscriber of OCS) and the OS operator, we will use a static agent (SA) that remains in the component of OS. The (SA) makes a *distant* CALL of *Interrupt(***in***:INC-CALL,***in***:BEFORE-CHECK-LOCS)* and then, as soon as an incoming call from a user $\mathcal{A}$ is interrupted in point *BEFORE-CHECK-LOCS*, the SA takes the following actions:

1. Make a *distant* CALL of *GetInfoIfSubscToOCS(***in***:*$\mathcal{A}$*,***out***:y)* to obtain the screening list of $\mathcal{A}$ (in y) if $\mathcal{A}$ is subscriber to OCS.
2. If $y = \emptyset$ (i.e., $\mathcal{A}$ is *not* subscriber to OCS): make a local CALL of *Resume(***in***:BEFORE-CHECK-LOCS)* to resume the previously interrupted incoming-call.
3. If $y \neq \emptyset$ (i.e., y contains $L_{ocs}^{\mathcal{A}}$): makes a local CALL of:
   - *Write(***in***:y,***in***:*L_{ocs}^{OS}*)* to obtain L_{ocs}^{OS}=$L_{ocs}^{\mathcal{A}}$
   - *Resume(***in***:BEFORE-CHECK-LOCS)* to resume the previously interrupted incoming call.

Discussions about the solutions of Sect. 3.2 and 3.2:
Note that the CS procedure (of Sect. 3.2) is obtained from the MA procedure (of Sect. 3.2) by removing items 1 and 3 and replacing the *local* CALL in Item 2 by a *distant* CALL towards the component of $\mathcal{A}$.

If several incoming calls are interrupted at the *same* time, then for *each* call, we have a MA (resp. SA) that takes the actions enumerated in Sect. 3.2 (resp. Sect. 3.2) and uses a particular L_{ocs}^{OS}. Therefore, we can have several resolutions executed in parallel.

In addition to the MA and CS resolution methods, the following third solution can be used: Every *Move* or distant CALL can be avoided if the incoming call from $\mathcal{A}$ contains the information that have to be returned by *GetInfoIfSubscToOCS(***in***:*$\mathcal{A}$*,***out***:y)*. This third solution is more performant than the other two solutions, in the sense that it reduces the number of transmitted messages. We will not consider it because it requires more complex processing in the side of callers.

4 Models of performance

Let us propose simple models of the cost and the delay to transmit an information I of size $\mathcal{S}_I$ from a node A to a node B in a network. More complex models can be found in the literature, but we think that our simple models are sufficient to achieve our goal, that consists of evaluating and comparing performances of two FI resolution methods. Our criteria of performance are:

- The cost $\mathcal{C} = cost_{A,B}(\mathcal{S}_I)$ for transmitting an information of size $\mathcal{S}_I$ from A to B.
- The delay (i.e., necessary time) $\mathcal{D} = delay_{A,B}(\mathcal{S}_I)$ for transmitting an information of size $\mathcal{S}_I$ from A to B.

Before to evaluate $\mathcal{C} = cost_{A,B}(\mathcal{S}_I)$ and $\mathcal{D} = delay_{A,B}(\mathcal{S}_I)$, we first need to evaluate the size of the transmitted information. When an information I of size $\mathcal{S}_I$ needs to be transmitted, some control information is added to it before the transmission. The size of total information that is transmitted, which we call *traffic*, can be modeled as follows:

$$\mathcal{T} = traffic(\mathcal{S}_I) = \Gamma_{\mathcal{S}_I}\mathcal{S}_I \tag{1}$$

$\Gamma_{\mathcal{S}_I}$ is called *overhead function* and we have $\Gamma_{\mathcal{S}_I} > 1$. Concretely, $(\Gamma_{\mathcal{S}_I} - 1)\mathcal{S}_I$ represents the quantity of control information that is added to I. This model is inspired from [9].

Definition 1 *(Packet) A* packet *is a message (including data and control information) that is sent in a single transmission.*

Notations 1 *($\mathcal{S}_d, \mathcal{S}_{ci}, \mathcal{S}_p$)*
$\mathcal{S}_d$ *denotes the maximum size of data in a transmitted packet,*
$\mathcal{S}_{ci}$ *denotes the size (usually constant) of control information in a packet,*
$\mathcal{S}_p$ *($= \mathcal{S}_d + \mathcal{S}_{ci}$) denotes the maximum size of a transmitted packet.*

Notations 2 *($a \div b$, a mod b) Let a and b be two integers:*
$a \div b$ *denotes the quotient of the division of a by b,*
a mod b *denotes the remainder of the division of a by b.*

The information I is sent in one or several packets; the number of packets is computed as follows:[1]

[1] For lack of space, proofs of several results will not be presented. In the sequel, these results will be underlined.

- $\mathcal{S}_I \div \mathcal{S}_d$ packets of size $\mathcal{S}_p$, plus
- 1 packet of size $(\mathcal{S}_I \bmod \mathcal{S}_d) + \mathcal{S}_{ci}$, if $(\mathcal{S}_I \bmod \mathcal{S}_d) \neq 0$.

We obtain the following overheads:

- If $(\mathcal{S}_I \bmod \mathcal{S}_d) \neq 0$ then: $\Gamma_{\mathcal{S}_I} = 1 + (1 + \mathcal{S}_I \div \mathcal{S}_d)\mathcal{S}_{ci}/\mathcal{S}_I$ (2)
- If $(\mathcal{S}_I \bmod \mathcal{S}_d) = 0$ then: $\Gamma_{\mathcal{S}_I} = 1 + (\mathcal{S}_I \div \mathcal{S}_d)\mathcal{S}_{ci}/\mathcal{S}_I$ (3)

The traffic is then computed by replacing, in (1), $\Gamma_{\mathcal{S}_I}$ by its expression (2) or (3). Before to model the cost and delay to transmit an information I from a node A to a node B of the network, here are some necessary notations:

Notations 3 *($\Lambda_{A,B}, \psi, \Omega_{A,B}, \Phi_{A,B}$)*
$\Lambda_{A,B}$ denotes the cost to transmit a packet of size 1 from A to B.
ψ denotes the necessary time to construct a packet of size 1.
$\Omega_{A,B}$ denotes the transmission rate (e.g., in bits per second) from A to B.
$\Phi_{A,B}$ denotes the propagation delay from A to B.

Model of cost: Let us evaluate $cost_{A,B}(\mathcal{S}_I)$ that denotes the cost for transmitting an information of size $\mathcal{S}_I$ from A to B. Assuming that the cost for transmitting a packet is proportional to $\mathcal{S}_I$ and $\Lambda_{A,B}$, we obtain:

$$\mathcal{C} = cost_{A,B}(\mathcal{S}_I) = \Lambda_{A,B}\mathcal{T} = \Lambda_{A,B}\Gamma_{\mathcal{S}_I}\mathcal{S}_I \quad (4)$$

The cost of traffic is evaluated by replacing in (4), $\Gamma_{\mathcal{S}_I}$ by its expression (2) or (3).

Model of delay: Let us evaluate $delay_{A,B}(\mathcal{S}_I)$ that denotes the necessary time for transmitting an information of size $\mathcal{S}_I$ from A to B.
The necessary time to construct all the packets is:

$$\psi\mathcal{T} = \psi\Gamma_{\mathcal{S}_I}\mathcal{S}_I \quad (5)$$

The necessary time to transmit all the packets is:

$$\frac{\mathcal{T}}{\Omega_{A,B}} + \Phi_{A,B} = \frac{\Gamma_{\mathcal{S}_I}\mathcal{S}_I}{\Omega_{A,B}} + \Phi_{A,B} \quad (6)$$

Therefore, the total time to construct and transmit all the packets is:

$$\mathcal{D} = delay_{A,B}(\mathcal{S}_I) = (\psi + \frac{1}{\Omega_{A,B}})\mathcal{T} + \Phi_{A,B} = (\psi + \frac{1}{\Omega_{A,B}})\Gamma_{\mathcal{S}_I}\mathcal{S}_I + \Phi_{A,B} \quad (7)$$

The delay of traffic is evaluated by replacing in (7), $\Gamma_{\mathcal{S}_I}$ by its expression (2) or (3).

5 Performance of OS-OCS resolution

Let us apply the models of Section 4 to evaluate and compare the performances of the two resolution methods of Sections 3.2 and 3.2. As we noted in Section 3.2, the only difference between the two methods is that:

MA-based method: a MA moves from OS to $\mathcal{A}$ (subscriber to OCS), makes a local CALL of *GetInfoIfSubscToOCS(***in***:*$\mathcal{A}$*,***out***:y)* and then returns to OS.
CS-based method: a SA in OS makes a distant CALL of *GetInfoIfSubscToOCS(***in***:*$\mathcal{A}$*,***out***:y)* towards $\mathcal{A}$.

The aim is that OS receives y, that contains the screening list of $\mathcal{A}$ if the latter is subscriber to OCS. Here are some necessary notations:

Notations 4 *($\mathcal{S}_{ma}, \mathcal{S}_{req}, \mathcal{S}_{rsp}, \mathcal{S}_{L_{ocs}}, \Lambda_{os,\mathcal{A}}, \Omega_{os,\mathcal{A}}, \Phi_{os,\mathcal{A}}$)*
$\mathcal{S}_{ma}$ *is the size of the MA when it moves from OS to* $\mathcal{A}$*;*
$\mathcal{S}_{req}$ *is the size of the request corresponding to the distant CALL from OS to* $\mathcal{A}$*;*
$\mathcal{S}_{rsp}$ *is the size of the response corresponding to the distant CALL when the screening list is empty;*
$\mathcal{S}_{L_{ocs}}$ *is the size of the screening list of* $\mathcal{A}$*, if any;*
$\Lambda_{os,\mathcal{A}}$ *is the cost to transmit a packet of size 1 between OCS and* $\mathcal{A}$*;*
$\Omega_{os,\mathcal{A}}$ *is the transmission rate between OS and* $\mathcal{A}$*;*
$\Phi_{os,\mathcal{A}}$ *denote the propagation delay between OS and* $\mathcal{A}$.

Control information is not considered in the definitions of $\mathcal{S}_{ma}, \mathcal{S}_{req}, \mathcal{S}_{rsp}$ and $\mathcal{S}_{L_{ocs}}$.

5.1 Performance of the MA resolution method

When the MA moves from OS to $\mathcal{A}$, its size is $\mathcal{S}_{ma}$. And when it moves back from $\mathcal{A}$ to OS, its size is $\mathcal{S}_{ma} + \mathcal{S}_{L_{ocs}}$. By using (1), the traffic generated is :

$$\mathcal{T}_{ma} = \Gamma_{\mathcal{S}_{ma}} \mathcal{S}_{ma} + \Gamma_{(\mathcal{S}_{ma}+\mathcal{S}_{L_{ocs}})}(\mathcal{S}_{ma} + \mathcal{S}_{L_{ocs}}) \tag{8}$$

where $\Gamma_{\mathcal{S}_{ma}}$ is determined by (2) or (3) for $\mathcal{S}_I = \mathcal{S}_{ma}$, and $\Gamma_{(\mathcal{S}_{ma}+\mathcal{S}_{L_{ocs}})}$ is determined by (2) or (3) for $\mathcal{S}_I = \mathcal{S}_{ma} + \mathcal{S}_{L_{ocs}}$. We can know evaluate the cost and delay induced by the traffic $\mathcal{T}_{ma}$ of (8) as follows:

Cost evaluation : By using (4), the cost induced by the traffic $\mathcal{T}_{ma}$ of (8) is:

$$\mathcal{C}_{ma} = \Lambda_{os,\mathcal{A}} \mathcal{T}_{ma} \tag{9}$$

Delay evaluation : By using (7), the delay induced by $\mathcal{T}_{ma}$ of (8) is:

$$\mathcal{D}_{ma} = (\psi + \frac{1}{\Omega_{os,\mathcal{A}}})\mathcal{T}_{ma} + \Phi_{os,\mathcal{A}} \tag{10}$$

5.2 Performance of the CS resolution method

When the SA makes a distant CALL from OS to $\mathcal{A}$, the size of its request is $\mathcal{S}_{req}$, and the size of the response is $\mathcal{S}_{rsp} + \mathcal{S}_{L_{ocs}}$. By using (1), the traffic generated is:

$$\mathcal{T}_{cs} = \Gamma_{\mathcal{S}_{req}} \mathcal{S}_{req} + \Gamma_{(\mathcal{S}_{rsp}+\mathcal{S}_{L_{ocs}})}(\mathcal{S}_{rsp} + \mathcal{S}_{L_{ocs}}) \tag{11}$$

where $\Gamma_{\mathcal{S}_{req}}$ is determined by (2) or (3) for $\mathcal{S}_I = \mathcal{S}_{req}$, and $\Gamma_{(\mathcal{S}_{rsp}+\mathcal{S}_{L_{ocs}})}$ is determined by (2) or (3) for $\mathcal{S}_I = \mathcal{S}_{rsp} + \mathcal{S}_{L_{ocs}}$. We can know evaluate the cost and delay induced by the traffic $\mathcal{T}_{cs}$ of (11) as follows:

Cost evaluation : By using (4), the cost induced by the traffic $\mathcal{T}_{cs}$ of (11) is:

$$\mathcal{C}_{cs} = \Lambda_{os,\mathcal{A}} \mathcal{T}_{cs} \tag{12}$$

Delay evaluation : By using (7), the delay induced by by $\mathcal{T}_{cs}$ of (11) is:

$$\mathcal{D}_{cs} = (\psi + \frac{1}{\Omega_{os,\mathcal{A}}})\mathcal{T}_{cs} + \Phi_{os,\mathcal{A}} \tag{13}$$

5.3 Performance comparison

Comparison of costs: The difference between the costs generated by the MA and CS resolution methods is determined by the following expression:

$$\mathcal{C}_{ma} - \mathcal{C}_{cs} \tag{14}$$

where $\mathcal{C}_{ma}$ and $\mathcal{C}_{cs}$ are determined by (9) and (12), respectively. We have the following results:

If $\mathcal{S}_{ma} - \mathcal{S}_{req} < 0$ **and** $\mathcal{S}_{ma} - \mathcal{S}_{rsp} < 0$ then (14) is negative, and thus, the MA method is less costly than the CS method.
If $\mathcal{S}_{ma} - \mathcal{S}_{req} > 0$ **and** $\mathcal{S}_{ma} - \mathcal{S}_{rsp} > 0$ then (14) is positive, and thus, the MA method is more costly than the CS method.
If $\mathcal{S}_{ma} - \mathcal{S}_{req} = 0$ **and** $\mathcal{S}_{ma} - \mathcal{S}_{rsp} = 0$ then (14) is null, and thus, the MA method is as costly as the CS method.

When all the overheads are (approximately) equal to 1, then (14) is equal to:

$$\Lambda_{os,\mathcal{A}}(2\mathcal{S}_{ma} - (\mathcal{S}_{req} + \mathcal{S}_{rsp})) \tag{15}$$

Therefore, if all the overheads are equal equal to 1, we have the following results:
If $2\mathcal{S}_{ma} - (\mathcal{S}_{req} + \mathcal{S}_{rsp}) < 0$, then MA method is less costly than CS.
If $2\mathcal{S}_{ma} - (\mathcal{S}_{req} + \mathcal{S}_{rsp}) > 0$, then MA is more costly than CS.
If $2\mathcal{S}_{ma} - (\mathcal{S}_{req} + \mathcal{S}_{rsp}) = 0$, then MA method is as costly as CS.

Comparison of delays: The difference between the delays generated by the MA and CS resolution methods is $\mathcal{D}_{ma} - \mathcal{D}_{cs}$. From (9), (12), (10) and (13), we obtain:

$$\mathcal{D}_{ma} - \mathcal{D}_{cs} = (\psi + \frac{1}{\Omega_{os,\mathcal{A}}})\frac{(\mathcal{C}_{ma} - \mathcal{C}_{cs})}{\Lambda_{os,\mathcal{A}}} \tag{16}$$

Since $(\psi + \frac{1}{\Omega_{os,\mathcal{A}}})$ and $\Lambda_{os,\mathcal{A}}$ are positive, we obtain the same results as for the difference of costs.

5.4 Interpretation: results, assumptions, and parameters

Recall that:

- the traffic generated by the MA solution is due to a *mobile agent* that moves from OS to the user of OCS and then moves back to OS;
- the traffic generated by the CS solution is due to:
  - a *distant request* sent from OS to the user of OCS, and
  - the *response* to the request.

Performance comparison between the two solutions is realized by comparing directly the traffics that they generate. More precisely, the solution that generates the smallest traffic also generates the smallest cost and delay.

Note that the assumption that the overheads are equal to 1 means that, in each link, the total size of control information is negligible w.r.t. the total size of data.

In our study, we have used the following parameters:

- $\Lambda_{os,\mathcal{A}}$ is a *commercial* parameter, that is, it is fixed by commercial criteria;
- $\Omega_{os,\mathcal{A}}$ and $\Phi_{os,\mathcal{A}}$ are *communication* parameters, that is, they depend on the communication environment;
- ψ, $\mathcal{S}_{ma}$, $\mathcal{S}_{req}$, and $\mathcal{S}_{rsp}$ are *computer* parameters, that is, they depend on the computer environment;
- $\mathcal{S}_{L_{ocs}}$ is theoretically unbounded.

Here is an example of values of the communication and computer parameters: $\Omega_{os,\mathcal{A}} = 1$ *Mbps*, $\Phi_{os,\mathcal{A}} = 100$ *ms*, $\psi = 1$ *ms*, $\mathcal{S}_{ma} = 150$ *bytes*, $\mathcal{S}_{req} = 100$ *bytes*, $\mathcal{S}_{rsp} = 200$ *bytes*.[2]

6 Conclusion and future work

The designers of telecommunications services are confronted with the following problem: services (or features) that behave correctly alone, may induce an undesired behavior once they are used together. This problem is commonly called *Feature Interactions* (FI). In this paper, we consider *distributed* FIs, that form

[2] *Mbps* denotes *Megabits per second*, *ms* denotes *milliseconds*.

the category of FIs the most difficult to solve because they involve services running in *different* components of the network. We propose simple models of performances to evaluate the cost and the delay of traffic in a network. Then, we apply these models to evaluate and compare performances of two methods, based on mobile agents and client-server technologies respectively, that solve a given FI.

Note that the applicability of the proposed models is general. For clarity we have used a simple example of FI, because the aim here was to introduce the principles of our approach of performance comparison.

In a near future, we intend to apply our study to more complex FIs, particularly those involving more than two services. We also plan to complete our study by using a specialized software tool, for example OPNET [10].

References

1. L.G. Bouma and H. Velthuijsen, editors. *Second International Workshop on Feature Interactions in Telecommunications Systems (FIW'94)*, Amsterdam, 8-10 May 1994. IOS Press.
2. K.E. Cheng and T. Ohta, editors. *Third International Workshop on Feature Interactions in Telecommunications Systems (FIW'95)*, Kyoto, 11-13 October 1995. IOS Press.
3. P. Dini, R. Boutaba, and Logrippo L, editors. *Fourth International Workshop on Feature Interactions in Telecommunications Systems (FIW'97)*, Montreal, 17-19 June 1997. IOS Press.
4. K. Kimbler and L.G. Bouma, editors. *Fifth International Workshop on Feature Interactions in Telecommunications and Software Systems (FIW'98)*, Lund (Sweden), 29 Sept.-1 Oct. 1998. IOS Press.
5. Muffy Calder and Evan H. Magill, editors. *Sixth International Workshop on Feature Interactions in Telecommunications and Software Systems (FIW'00)*, Glasgow (Scotland, UK), 17-19 May 2000. IOS Press.
6. D. Amyot and L. Logrippo, editors. *Seventh International Workshop on Feature Interactions in Telecommunications and Software Systems (FIW'03)*, Ottawa (Canada), 11-13 June 2003. IOS Press (Netherlands).
7. S. Reiff-Marganiec and M. D. Ryan, editors. *Eighth International Workshop on Feature Interactions in Telecommunications and Software Systems (FIW'05)*, Leicester (UK), 28-30 June 2005. IOS Press (Netherlands).
8. A. Khoumsi and S. Cherkaoui. Use of agents for resolving feature interactions. In *Proc. 3rd Intern. Workshop on Mobile Agents for Telecommunications Applications (MATA)*, Montreal, Quebec, Canada, August 2001.
9. M. Baldi and G. P. Picco. Evaluating the tradeoffs of mobile code design paradigms in network management applications. In *proc. 20th International Conf. on Software Engineering (ICSE)*, April 1998.
10. http://www.gel.usherb.ca/khoumsi/Opnet/index.html.

Guided Local search for the Examination Timetabling Problem

Drifa Hadjidj[1] and Habiba Drias[2]

1Faculté des sciences, Université M'hamed Bouguerra, Boumerdes, Algérie.
dhadjidj@hotmail.com

[2]Institut National d'Informatique–INI- BP 68M OUED SMAR Alger Algérie
drias@wissal.dz

Abstract: Examination timetabling problem is an optimization problem which consists in assigning a set of exams to a set of contiguous time slots, satisfying a set of constraints. The problem falls in the category of the NP-Complete problems and is usually tackled using heuristic methods. In this paper we describe a solution algorithm and its implementation based on the graph heuristics and the guided local search. The implementation of the algorithm has been experimented on the popular Carter's benchmarks and compared with the best recent results.

Keywords: Examination timetabling, guided local search, meta-heuristic, graph heuristics.

1 Introduction

A general timetabling problem includes assigning a set of events (courses, exams, meetings, etc) into a limited number of timeslots (periods),

All timetabling problems share the same basic characteristics of the general timetable problem. But, the amount of variance between different timetable problems is so greet as to make general solutions impracticable to implement. In our study we are mainly concerned with one type of university timetabling problem: the examination timetabling problem.

This problem regards the scheduling for a number of exams in a given set of periods avoiding overlap of exams with common students and spreading the exams for student as much as possible. However, as polices differ from university to university in defining the quality of exam timetabling; it becomes difficult to give a universal definition of this problem.

Note that the examination timetabling problem is an NP-hard problem [25] which means that it is not possible in a reasonable time to carry out an exhaustive search for the timetable.

The Manuel solution of timetabling usually requires several days and may be unsatisfactory to some respects.

Many of the exam timetabling algorithms which have been developed are very problem-specific and cannot be used to find high quality solutions to a wide range of

problem instances. In general, no one algorithm will give best results for all exam timetabling problems to which it is applied. Some algorithms will perform well on one subset of problems but fail when different constraints are introduced

A large number of variants of the examination timetabling problem have been proposed in the literature depending on the type of the institution and distinct constraints. In other respects, several techniques have been presented to solve each type of timetabling problem.

The early approaches include *integer programming*, *network flow*, *and Graph heuristics* [22]. Some of these techniques are either impractical or too simple to solve complex or large timetabling problems.

Constraint programming logic [7] has been employed over the years for timetabling problems.

*Heuristic sequencing approach [*19, 22], employed in educational timetabling, involves using heuristic, to estimate how difficult each exam will be to place in timetable and to order them before placing each exam. These heuristics produce substantially better timetables than using a random ordering.

Recently a large amount of successful research has investigated *meta-heuristic* approaches for a variety of timetabling problems. These include *taboo search simulated annealing and, genetic algorithms* [23, 30].These methods begin with one or more initial timetables and employ search strategies which try to avoid local optima. Extensive work has also been carried out to study other new approaches and methodologies for timetabling problems as more problem solving experience is collected. These include *Hyper-heuristics* [17], *Hybrid methods* [27], *Cluster methods* [4], *Case-based reasoning* [13], *hybrid multi-objective evolutionary approach* [21], *Ant algorithms* [24], *Sequential methods with meta-heuristics* [15], *Fuzzy methodologies* [5]*, Multicriteria approach*, [9], and *Large Neighbourhood Search Approach* [1, 2, and 3].

Note that in this paper we are not attempting to directly solve the examination timetabling problem. Instead, we are concerned with underpinning the development of general timetabling systems by investigating guided local search approach for this problem. Thus, we describe a solution algorithm and its implementation based on the meta-heuristic called guided local search. The implementation of the algorithm has been experimented on the popular Carter's benchmarks and compared with the best recent results.

The paper starts with the description of the exam timetabling problem, followed by the general template of Guided Local Search meta-heuristic. In section 4, we present our algorithm for the examination timetabling based the Guided Local Search meta-heuristic followed by the computational experimentation with the performances obtained, before concluding our contribution on the field.

2 Description Problem

The examination timetabling problem can be described as a given set of *P* periods and a set of *N* exams to place each exam in some period avoiding overlap of exams with common students and spreading the exams for student as much as possible. Here we

assume that all exams have the same duration period and we give a version based on [15, 19, and 30] to allow comparison of quality of solutions produced by the best recent approaches.

It is common in timetabling community to employ the penalty function for solution evaluation proposed by Carter et al. [19] on his wide range of real world timetabling benchmark problems, available via internet: *ftp://ftp.mie.utoronto.ca/pub/carter/tesprob/.*

This penalty function enables comparison of quality of solutions produced by different approaches. Thus in our experiment we use the method of solution evaluation g(t) (shown in expression (1) for measuring student spread based on proximity to minimize. The hard constraints that must be observed are presented in expressions (2) and (3). The expression (2), stats that every event should be scheduled once, and only once in the timetable. The expression (3) specifies that no conflicting exams should be scheduled within the same period. As in [19] we assume that we have not a limitation on available seating for each period.

The penalty function sums all the penalties and divides the obtained sum by the number of students. The final result is the average penalty for each student. The smaller its value the better the solution is. Note that whenever a student has to sit two examinations *s* periods apart, such penalties are: $w_1 = 16$, $w_2 = 8$, $w_3 = 4$, $w_4 = 2$, $w_5 \; w5 = 1$.

Thus the notation used is as follows:

$t_{ip} = 1$ If exam *i* is scheduled in period *p*, 0 otherwise

c_{ij} The number of students attending exam *i* and *j*

M The total number of students

The minimization problem is defined as minimize:

$$(\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\sum_{p=1}^{P}\sum_{q=1}^{P} 2^{5-|q-p|}\, t_{ip}\, t_{jq}\, c_{ij} + \sum_{i=1}^{N} 5000\, t_{i(P+!)}) / M \qquad \textbf{(1)}$$

If $1 \le |q-p| \le 5$ and 0 otherwise

$$\text{Subject to } \sum_{p=1}^{P+1} t_{ip} = 1 \qquad \forall i \in \{1,.....N\} \qquad \textbf{(2)}$$

$$\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}\sum_{p=1}^{P} t_{ip}\, t_{jp}\, c_{ij} = 0 \qquad \textbf{(3)}$$

As in some cases it may not be possible to construct a feasible timetable, each exam that cannot be scheduled in a valid period can be placed in an extra period, the period *(P+1)*. We have no hard constraints to prevent scheduling in this period, but it should be heavily penalized [8, 14].

3 Guided Local Search

One way to contain the combinatorial explosion problem of many real life constraint optimization problems which are hard to solve using complete methods is to sacrifice completeness. Some of the best known methods that use this strategy are local search methods. Starting from a candidate solution, which may be randomly or heuristically generated, the search moves to a neighbor which is 'better' according to the objective function (lower in a minimization problem and higher in a maximization problem). The search terminates if no better neighbor can be found, or resources run out. The whole process can be repeated from different starting points.

One of the main problems with each one of these methods is that it may settle in local optima states which are better than all their neighbors but not necessarily the best possible solution. To overcome that, and escape local optima, methods such as *Simulated Annealing* and *Tabu Search* [23, 26, 30] have been proposed.

Guided Local Search (GLS) originally proposed by Voudouris and Tsang [33] is also a suitable meta-heuristic for a wide range of large and difficult combinatorial optimization problems. Successful applications of this method so far include practical problems such as Frequency Allocation [33], Workforce Scheduling [31], routing and scheduling problems [33] and also classic problems such as the Travelling Salesman Problem (TSP) and the Quadratic Assignment Problem (QAP) [31], [34].

The main focus of GLS is on the exploitation of problem and search-related information to effectively guide local search heuristics in the vast search spaces of NP-hard optimization problems. This is achieved by augmenting the objective function of the problem to be minimized, for example, with a set of penalty terms based on particular solution features not considered to be part of a near-optimal solution which are dynamically manipulated during the search process to steer the heuristic to be guided.

The more often a feature appears and is penalized, the less likely it is to be penalized further. GLS associates a cost and penalty to each feature. The cost refers often to the values of the original objective function, and the penalties are initially set to 0 and are updated during the search each time a local minimum is reached.

4 Guided Local Search for the Examination Timetabling Problem

In this section, we describe the solution approach that we have developed for the examination timetabling problem using GLS approach.

Given our objective function g which maps every candidate solution *t* to a numerical value, we define a function *h* which will be used by the local search algorithm replacing *g*.

$$h(t) = g(t) + \lambda \cdot \sum_{i=1}^{K} p_i \cdot I_i(t) \quad (4)$$

Where t is a candidate solution, λ is a parameter to the GLS algorithm, K *is* the number of features,

p_i is the penalty for feature i (which are initialized to 0) and I_i is an indication of whether the candidate solution t exhibits feature i*:*

$I_i(t) = 1$) if t exhibits feature i; 0 otherwise.

Features are unwanted characteristics of a solution or in our case unscheduled exams.

When the local search settles on a local optimum, the penalty of some of the features associated to this local optimum is increased; this has the effect of changing the objective function and driving the search towards other candidate solutions. Another factor which should be considered is the current penalty value of that feature. We define the utility of penalizing feature i, under a local optimum t *, as follows:

$$util(t^*, f_i) = I_i(t^*) \cdot \frac{c_i}{1 + p_i} \quad (5)$$

If a feature is not exhibited in the local optimum, then the utility of penalizing it is 0. The higher the cost of this feature, the greater the utility of penalizing it. Besides, the more times that it has been penalized, the greater $(1 + p_i)$ becomes, and therefore, the lower the utility of penalizing it again. In a local optimum, the feature(s) with the greatest $util$ value will be penalized by adding 1 to its penalty value:

$$p_i = p_i + 1 \quad (6)$$

The λ parameter in the GLS algorithm is used to adjust the weight of penalty values in the objective function.

When we take into consideration cost and the current penalty in selecting the feature to penalize, we distribute the search effort in the search space. Candidate solutions which exhibit features involving lower cost, are given more effort in the search, but penalties may lead the search to explore candidate solutions which exhibit features with higher cost.

An outline of the meta-heuristic and the associated local search Strategy is described in Algorithm 1.

Algorithm 1. GLS for the Examination timetabling Problem

---

1) Generate a starting candidate solution heuristically using the Generation Method presented in 4.1

2) Initialize all the penalty values p_i to 0;
3) Repeat the following until a termination condition (e.g. a maximum number of iterations or a time limit) has been reached:
   - Perform local Search Strategy described in 4.2 according to the function *h* (which is *g* plus the penalty values, as defined in expression (4) until a local optimum *t** has been reached;
   - For each feature *i* which is exhibited in *t** compute $util_i = c_i /(1 + p_i)$;
   - Penalize every feature *i* such that $util_i$ is maximum: $p_i = p_i + 1$;
4) Return the best candidate solution found so far according to the objective function *g*.

4.1 Generation Method

In order to start with a good initial solution, we combine heuristics with the process of tournament selection. First, we generate a subset of exams randomly, we order them with respect to some heuristic, and then we assign them one by one into feasible timeslots without violating any hard constraint and with the lowest penalty. The size of the subset is given as percentage of the full set of examinations [15, 19].

A variety of sequential heuristics can be used to construct initial solutions. They sort examinations on the basis of the estimated difficulty of their scheduling. The most interesting are: *Largest Degree heuristic and Saturation Degree heuristic* which are two widely studied graph coloring heuristics for applications to timetabling problems [6, 15, and 19], see Algorithm 2.

In *Largest Degree heuristic (LD)* Exams are ordered decreasingly by the number of conflicts they have with other exams. This heuristic aims to schedule the most conflicting exams first.

For *the Saturation Degree heuristic (SD)* Exams that are not yet scheduled are ordered increasingly by the number of feasible timeslots available at that time. The priorities of the exams thus change dynamically according to the situations encountered at each step of the solution construction

Algorithm 2: generation Method

- Randomly built a tournament of exams with size at a set proportion of total size (for example 20% of total size.
- Select in turn an exam from tournament using one of the precedent ordering strategies (to break ties we use the *LD* strategy)

- Schedule the selected exam in the valid period causing least penalty. If there is a tie between periods schedule the exam at the earliest of the periods. If no such period exists then add to the list of unscheduled exams.

4.2 local Search Strategy

This local search algorithm iteratively inspects the neighborhood (the set of candidate solutions, produced from the current one by moving one exam) and replaces the current solution by the candidate with lowest penalty. The hill climbing used to improve the solutions does not accept zero improvement and will not spend time exploring regions of no immediate improvement such as plateaus. Algorithm 3 can be summarized as follows:

Algorithm 3: Improvement Method

- For each period p in the timetable taken in some random order
- Take each exam *e* in period *p* in some random order.
- Calculate the penalty of scheduling e in every other valid period
- If the best of these has low penalty than *p*, then schedule *e* in this period. Using the precedent ordering strategy *SD,* try scheduling each unscheduled exam in some valid period
- If any improvement is found repeat

---

5 Experimental Experiences

The purpose of this experiment is to evaluate the performance of our approach for the examination timetabling problem.

We assume that there are no limits on the amount of seating available during each period, in order to enable comparison with a wider range of existing methods based on this assumption. We implement and run the algorithm using Visual C++ 6, on a Pentium III 851 MHz machine with 256 Mb RAM.

The algorithm was tested on 5 real world benchmark exam timetabling problems using the Carter *et al.* (1996) data collection [19]. These datasets presented in Table 1, cover a range of characteristics (on number of exams and conflict matrix density).

Table 1. Characteristics of the problems used for testing

Datasets	Nb. of Exams	Nb. of Students	Nb. of Periods	density
HEC-S-92	81	2823	18	0.42
STA-F-83	139	611	13	0.14

UTA-S-92	622	21266	35	0.13
UTE-S-92	184	2750	10	0.08
EAR-F-83	190	1125	24	0.27

In Table 2, we show the performance of our approach in comparison with selected recently published results on the datasets. The best results amongst the compared techniques for each dataset are highlighted in bold font.

Table 2. The objective function results

Datasets	ute92	uta92	sta83	hec92	ear83
Our GLS approach	26.94	3.5	157.12	12.60	35.90
Abdullah et al. (2006) [1]	**24.21**	3.63	150.28	10.28	34.87
Abdullah et al. (2006) [3]	26	3.6	159	10.8	36
Asmuni et al. (2004)[5]	27.78	3.75	160.42	11.78	37.02
Burke et al. (2004)[8]	25.7	3.5	159.1	10.6	35
Burke et al. (2005) [10]	35.40	4.52	158.2	-	45.60
Burke e t al. (2007) [11]	28.01	**2.88**	158.19	12.25	37.92
Burke e t al. (2002) [12]	25.83	3.2	168.73	11.54	37.05
Burke et al. (2006) [16]	24.82	3.06	157.03	10.15	32.76
Caramia et al. (2001) [18]	24.4	3.5	158.2	**9.2**	**29.3**
Carter et al. (1996) [19]	25.8	3.5	161.5	10.8	36.4
Casey et al. (2002) [20]	25.4	-	**134.9**	10.8	34.8
Cote et al. (2005) [21]	25.3	3.5	157	10.4	34.2
Di Gaspero et al. (2001) [23]	29	4.2	160.8	12.4	45.7
Eley et al. (2006) [24]	27.7	3.8	157.2	11. 3	36.8
Kendall et al. (2005)[28]	27.12	3.77	157.38	10.74	38.14
Merlot et al. (2003) [29]	25.1	3.5	157.3	10.6	35.1
Petrovic et al. (2003) [30]	25.33	3.14	165.27	10.83	33.75
White et al. (2004) [37]	29	4.44	158	12.9	45.8
Yang et al. (2004) [38]	25.53	3.14	151.15	10.83	33.7

As the selection of the search parameters has an important effect on the performance of the meta-heuristic, computational analyses are conducted and these parameters are fixed as follows:

The size of the subset in the process of tournament selection = 20

The value of λ =50

We can observe that the performance of this approach with these parameters is comparable and competitive in comparison to the others. Another advantage of this approach is that it cans enhance the population based meta-heuristics to obtain a number of high-quality solutions from which the decision-maker could choose the one to use.

6 Conclusion

In this paper we have developed and implemented an approach based on the Guided Local Search meta-heuristic to the examination timetabling. The results by this method enhanced by the use of the most performing graph heuristics are encouraging. We plant to improve our work by considering other soft constraints and hybridizing this approach with other metaheuristics.

References

1. Abdullah, S. et al.: Investigating Ahuja-Orlin's Large Neighbourhood Search Approach for Examination Timetabling. Accepted for publication in OR Spectrum, 2006.
2. Abdullah, S. et al.: A Tabu based Large Neighbourhood Search Methodology for the Capacitated Examination Timetabling Problem. Accepted for publication in the Journal of Operational Research Society, 2006.
3. Abdullah, S. et al., A Multi-start Large Neighborhood Search Approach with Local Search Methods for Examination Timetabling. In: The International Conference on Automated Planning and Scheduling (ICAPS 2006) Editors: Derek Long, Stephen F. Smith, Daniel Borrajo and Lee McCluskey. Cumbria, UK, 6-10, pp 334-337, June 2006.
4. Arani, T. et al.,A Three Phased Approach to Final Exam Scheduling", IIE Trans. 21, pp. 86-96, 1989.
5. Asmuni H., et al..,: Fuzzy Multiple Ordering Criteria for Examination Timetabling. Selected papers from the 5th International Conference on the Practice and Theory of Automated Timetabling. Pittsburgh, USA. Aug 2004. Springer Lecture Notes in Computer Science, vol. 3616, pp. 334-353, 2004.
6. Brelaz. D.: new methods to colour the vertices of a graph. Communications of the Association for Computing Machinery, 22, pp.251–256, 1979.
7. Boizumault, P., et al.: Constraint logic programming for examination timetabling. Journal of Logic Programming, 26(2), pp.217–233, 1996.
8. Burke, E.K., et al.: A time-predefined local search approach to exam timetabling problems. IIE Transactions on Operations Engineering 36, pp.509- 528, 2004.
9. Burke, E.K., et al.: A Multicriteria Approach to Examination Timetabling. Lecture Notes in Computer Science vol. 2079. The Practice and Theory of Automated Timetabling III: Selected Papers (PATAT 2000). Burke, E.K., Erben, W. (Eds), Springer-Verlag, Berlin, Heidelberg, New York, pp. 118-131, 2001.
10. Burke, E., et al.: Hybrid Graph Heuristics within a Hyper-heuristic Approach to Exam Timetabling Problems B.L. Golden, S. Raghavan and E.A. Wasil (Eds.). The Next Wave in Computing, Optimization, and Decision Technologies. Springer. pp. 79-91, 2005.
11. Burke, E.K., et al.: A Graph-Based Hyper Heuristic for Educational Timetabling Problems. To appear in European Journal of Operational Research, 176, pp 177-192, 2007.
12. Burke E.K.et al.: Enhancing Timetable Solutions with Local Search Methods. In: Burke E.K. and De Causmaecker P. (eds.) Selected Papers from the 4 the International Conference on the Practice and Theory of Automated Timetabling, Lecture Notes in Computer Science 2740. pp. 195-206, 2002
13. Burke, E.K., et al.: Case Based Heuristic Selection for Timetabling Problems. Journal of Scheduling, vol 9, no. 2, pp. 115-132, 2006.
14. Burke, E.K., et al.: Solving examination timetabling problems through adaption of heuristic ordering. Annals of Operations Research, 129 pp. 107-134, 2004.

15. Burke, E. K.et al. :A Simple Heuristically Guided Search for the Timetable Problem, Proceedings of the International ICSC Symposium on Engineering of Intelligent Systems (EIS'98), E. Alpaydin, C Fyte (eds), university of La Laguna, Spain, published by ICSC Academic Press, , pp. 574-579, 1998.
16. Burke, E. K.et al.: Solving Exam Timetabling Problems with the Flex-Deluge Algorithm. Proceedings of The 6 th International Conference on the Practice and Theory of Automated Timetabling 30 th August – 1 st September 2006, pp 167-180, 2006.
17. Burke E., et al.: A Tabu Search Hyperheuristic for Timetabling and Rostering. Journal of Heuristics. 9(6), pp 451-470, 2003.
18. Caramia, M., et al.: New Algorithms for Examinations Timetabling. In Naher, S., Wagner, D., editors, Algorithm Engineering 4th Int. Workshop, Proc. WAE 2000 (Saarbrucken, Germany, September. Lecture Notes in Computer Science, Springer-Verlag, Berlin Heidelberg New York. Vol. 1982, pp.230-241, 2001.
19. Carter, M.W.,G.et al., :Examination timetabling: Algorithmic strategies and applications. Journal of the Operational Research Society. 47, pp 373-383, 1996.
20. Casey, S. et al.: A Hybrid Algorithm for the Examination Timetabling Problem. Practice and Theory of Automated Timetabling IV (PATAT 2002, Gent Belgium, August, selected papers). Lecture Notes in Computer Science, Vol. 2740. Springer-Verlag, Berlin Heidelberg New York 6, pp. 205-230, 2002.
21. Côté, P., et al.: A hybrid multi-objective evolutionary algorithm for the uncapacitated exam proximity problem. In: Practice and Theory of Automated Timetabling V, E. K. Burke pp 294-312, 2005.
22. De Werra, D. "An introduction to timetabling problem". European Journal of Operational Research, 19, pp. 151 – 162. , 1985.
23. Di Gaspero, L. and Schaerf, A.: Tabu Search Techniques for Examination Timetabling. Lecture Notes in Computer Science vol. 2079. The Practice and Theory of Automated Timetabling III: Selected Papers (PATAT 2000). Burke, E.K., Erben, W. (Eds), Springer-Verlag, Berlin, Heidelberg, New York, pp. 104-117. , 2001.
24. Eley M. :Ant algorithms for the exam timetabling problem Proceedings of The 6 th International Conference on the Practice and Theory of Automated Timetabling 30 th August – 1 st September, pp 167-180, 2006.
25. Karp. R. M. : :Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, Complexity of Computer Computations, Plenum Press, New York, pp. 85–103, 1972.
26. Kendall, G.et al.: An Investigation of a tabu search based hyperheuristic for examination timetabling. In: Kendall G., Burke E. and Petrovic S.(eds), Selected papers from Multidisciplinary Scheduling; Theory and Applications, pp 309-328, 2005.
27. Merlot, L.T.G., et al.: A hybrid algorithm for examination timetabling problem. In: Burke, E, Causmaecker, P.D. (Eds.): Practice and Theory of Automated Timetabling IV (PATAT 2002, Gent Belgium, August, selected papers). Lecture Notes in Computer Science, Vol. 2740. Springer-Verlag, Berlin Heidelberg New York pp.207-231. 2003
28. Petrovic, S., et al.: Case-based Initialisation for Examination Timetabling. Proc. of 1st Multidisciplinary Intl. Conf. on Scheduling: Theory and Applications (MISTA 2003), Nottingham, UK, Aug 13-16, pp. 137-154, 2003.
29. Schaerf, A.: A Survey of Automated Timetabling". Artificial Intelligent Review, 13, pp. 87-127, 1999.
30. Thompson et al.: A robust simulated annealing based examination timetabling system. Computers and Operations Research, 25, pp 637--648, 1998.
31. Tsang, E., and Voudouris, C.: Fast local search and guided local search and their application to British Telecom's workforce scheduling problem, Operations Research Letters, Vol. 20, No. 3, 119-127, 1997.

32. Voudouris C.:Guided local search for combinatorial problems.PhD dissertation,University of Essex,UK,1997.
33. Voudouris C,Tsang E.:Guided local search.European Journal of Operations Research, 113:80–119, 1998.
34. Voudouris,C.,Tsang,E.,:.Guided local search and its application to the traveling salesman problem.European Journal of Operational Research, 113,469 –499, 1999.
35. White, G.M., et al.: "Using tabu search with longer-term memory and relaxation to create examination timetables", European Journal of Operational Research, 153(16), pp.80-91, 2004.
36. Yang S. et al.: A Novel Similarity Measure for Heuristic Selection in Examination Timetabling. in E.K. Burke and M. Trick, editors: The Practice and Theory of Automated Timetabling V: Proceedings of the Fifth International Series of Conferences on the Practice and Theory of Automated Timetabling, Pittsburgh, USA, pp 377-396, 2004.

Une approche infométrique pour l'analyse de données bibliographiques

Bilal YALAOUI [1], Hakim HARIK [2]

CERIST, Centre de recherche sur l'information scientifique et technique, Rue des 3 Frères AISSOU, Ben-Aknoun BP143, Alger 16030
[1] yalaoui@cerist.dz, [2]hhakim@cerist.dz

Résumé. L'infométrie est un ensemble d'activités métriques relatives à l'information pour analyser la littérature scientifique et technique produite par les chercheurs, les inventeurs, les ingénieurs, alors que le clustering, appelé également classification non-supervisée, est un processus permettant d'organiser un ensemble de données en classes cohérentes ou homogènes, appelées clusters. Dans cet article, on s'intéresse à une analyse infométrique de données bibliographique basée sur une technique de clustering pour classer et représenter graphiquement une masse d'information bibliographique issues d'une requête donnée par un utilisateur.

Mot-clés. Base de données bibliographiques, importance d'une arête, infométrie, corpus textuel, graphe d'association de termes,

1 Introduction

Les bases de données bibliographiques sont un outil fondamental de la société de l'information. Depuis leurs introductions, un grand nombre de telles bases de données dans beaucoup de domaines ont été développées. Tout au long des années, les développeurs ont cherché des manières de suivre le progrès de la technologie, de contrôler de vastes quantités de données, de traiter le placement limité et de répondre à des espérances toujours croissantes d'utilisateur. Par conséquent, le disque compact, en ligne et les versions de Web avec les interfaces utilisateur amicales ont été développés (Medline, ISI's databases, ERIC, PsycINFO etc.).

De ces faits la maîtrise du processus d'analyse et d'accès à l'information dans les bases de données bibliographiques demeure un problématique et un enjeu majeur pour les scientifiques et les professionnels de l'information. Actuellement, dans les centres de recherches scientifiques publics et privés il existe un intérêt grandissant à l'analyse des informations contenues dans les bases de données. Cet intérêt se traduit par l'utilisation de cette mine d'informations à des fins stratégique : la gestion stratégique de la recherche et de la technologie, la détermination d'une politique de scientifique, la surveillance d'un domaine scientifique avec ses multiples aspects (la synthèse

d'information, l'observation des tendances, le positionnement des acteurs de la recherche, etc.…).

Dans notre contribution, on s'intéresse à une approche infométrique pour l'analyse de l'information scientifique et technique dans des notices bibliographiques qui comprend une analyse de contenu à partir des résumés et titres sous forme graphique.

2 Bases théoriques de l'approche

2.1 Analyse infométrique de données bibliographiques

L'analyse infométrique de l'information scientifique et technique dans des bases de données bibliographiques comprend une analyse de contenu à partir des mots-clés, résumés et titres pour analyser la littérature scientifique et technique produite par les chercheurs, les inventeurs et ingénieurs ce qui permet :

- évaluer un laboratoire,
- évaluer le fonds de périodiques d'une bibliothèque
- suivre l'évolution d'un thème de recherche
- apprécier l'impact d'un article
- évaluer la qualité d'une revue

Les méthodes d'analyse infométrique connus dans la littérature scientifique peuvent être divisés en deux classes :

- Uni-dimensionnelles : Etude des publications (articles, brevets) et leur contenu, en se basant sur des comptages statistiques des publications ou d'éléments extraits de ces publications par les lois de distributions bibliométriques [1], [5], [8] ce qui permettent de quantifier un domaine ou un thème de recherche ou une équipe par sa magnitude (nombre d'articles, nombre de revues), son actualité (selon la date de publication), sa localisation (selon le pays d'édition des revues scientifiques), l'importance des périodiques scientifiques (nombre d'articles dont ils sont la source au cours d'une période déterminée) et la localisation des auteurs (selon leur appartenance institutionnelle).

- Méthodes relationnelles : Existence de liens entre les publications (cocitation, mots associés, ...). Le principe de base est de regrouper les « objets » similaires du point de vue de leurs caractéristiques.

Dans notre cas, on s'intéresse aux méthodes relationnelles par une technique de clustering d'un graphe.

2.2 Le clustering

Un clustering, appelé également classification non-supervisée [3], est un processus qui permet d'organiser un ensemble de données en classes cohérentes ou homogènes, appelées clusters, pour simplifier la représentation des données initiales et qui s'applique, a priori, sur n'importe quel type de données.

Le problème du clustering se ramène, pour l'essentiel, à un problème de partitionnement du graphe [6], [7]. Un partitionnement est la division des données ou

objets en groupes. Chaque groupe, appelé communauté (cluster), est un ensemble d'objets partageant un intérêt commun qui sont similaires entre eux et dissimilaires aux objets des autres groupes. En terme de graphe, elle consiste à regrouper les sommets en communautés selon un critère prédéfinit.

Une communauté dans un graphe correspond intuitivement à l'existence de groupes de sommets plus fortement connectés entre eux que vers les autres sommets. Les communautés peuvent avoir des interprétations différentes suivant le type de réseau à considérer. Le terme de communauté dans un réseau d'information a souvent été synonyme de « thème ». Ainsi, dans ce réseau la problématique revient à organiser un ensemble de données en différentes thématiques.

Pour ce faire, un corpus de notices bibliographique noté Ψ qui constitue la base de notre étude à partir d'une requête donnée doit être formée.

Ainsi, une interrogation est réalisé à l'aide d'une requête *i*, une fois la recherche lancée, on récupère les résultats à partir de la base de données bibliographiques utilisée, ce qui réalise une base de donnée locale formé d'un ensemble de *n* références bibliographiques constitue sur le sujet étudié comme cela est illustré sur la figure 1 :

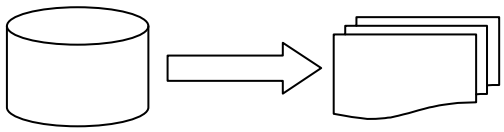


Figure 1.

A partir du corpusΨ, on extrait, par un outil d'extraction de termes, un ensemble de termes significatifs dans les champs titres et résumés des notices bibliographiques pour construire le graphe d'association de termes (voir paragraphe suivant) afin de le partitionner en communautés par la notion d'importance d'une arête.

2.3 Le modèle de graphe d'association de termes

Pour mettre en évidence les relations qui existent entre les éléments d'un ensemble $\mathcal{T}$ de termes significatifs extraites à partir d'un corpus donné Ψ, nous proposons d'utiliser le modèle de graphe d'association de termes.

Un graphe d'association de termes d'un corpus est un graphe dans lequel il montre toutes les relations qui existent entre les termes d'un ensemble $\mathcal{T}$, qui est extrait à partir d'un logiciel d'extraction de termes, sous la forme d'un graphe. Ce type de graphe considère les termes de $\mathcal{T}$ comme des indicateurs de connaissance (contenu des notices bibliographiques) et se base sur leurs cooccurrences pour mettre en évidence la structure de leurs relations. L'idée de cooccurrence est essentielle. En

effet, si on considère que deux notices sont proches parce qu'ils sont indexés par des termes similaires, alors deux termes figurant ensemble dans un grand nombre de notices seront considérés comme proches. Cependant, la cooccurrence ne permet pas à elle seule de mesurer la force des associations entre termes (leur proximité), car elle avantage les termes de haute fréquence par rapport à ceux de basse fréquence. L'emploi d'un indice statistique approprié permet de normaliser la mesure de l'association entre deux mots-clés. En pratique, nous utilisons le plus souvent l'indice d'Équivalence dont les valeurs varient entre 0 et 1.

De manière structurelle, nous posons:

H = {c / c est une notice bibliographique du corpus Ψ },

Soit $\mathcal{T}$ un ensemble de termes du corpus textuel Ψ:

$\mathcal{T}$ = {i / i est un terme dans le corpusΨ qui est extrait par un logiciel d'extraction de termes }

Pour chaque terme $i \in \mathcal{T}$, on pose :

$$D(i) = \{ c \in H / i \text{ un terme de } c \}$$

$D(i)$ représente l'ensemble de tous les notices bibliographique contenants le terme i et donc le nombre d'occurrences t_i d'un terme i dans un corpus Ψ correspond au nombre de notices bibliographiques contenants le terme i dans ce corpus alors que le nombre de cooccurrences t_{ij} représente le nombre d'apparition commune d'un terme i avec un autre terme j c'est-à-dire le comptage du nombre de fois de la présence simultanée de deux termes dans le corpus, ce qui permet de rechercher les associations de termes fréquents.

La nombre d'occurrences et le nombre cooccurrences peuvent être exprimés en fonction de $D(i)$, ainsi :

- $t_i = |D(i)|$.
- $t_{ij} = |D(i) \cap D(j)|$.

Définition.1. Soit $\mathcal{T}$ un ensemble de termes significatifs d'un corpus Ψ extrait à partir d'un logiciel d'extraction de termes.

Un graphe d'association de termes $G = (V, E, W)$, relative au corpus Ψ, est un graphe simple non orienté et pondéré, tel que :

- $V = \mathcal{T}$.
- $\mathcal{W}$ fonction degré d'association de termes définie de $V \times V$ vers [0, 1] telle que :

$\forall i, j \in V : W(i, j) = \dfrac{t_{ij}^{\,2}}{t_i \times t_j}$ notée w_{ij}

- $\forall i \in V, \forall j \in V : ij \in E \Leftrightarrow w_{ij} > 0 \Leftrightarrow D(i) \cap D(j) \neq \phi$.

Ce graphe permet de montrer l'influence entre un terme et son co-occurrent.

3 La notion d'importance d'une arête

3.1 Le principe

Le principe de partitionnement du graphe est basé sur la connectivité des voisinages d'une arête pondérée. En d'autre terme, elle se base sur la densité de liens qui existe entre les sommets proches qui est représenté dans notre cas par la densité de liens dans le voisinage des extrémités d'une arête.

Considérons En effet, une communauté dans un graphe correspond intuitivement à l'existence des sommets fortement connectés entre eux. Alors si les sommets du voisinage appartiennent à la même communauté les liens entre ces sommets deviennent importants donc les deux voisinages des extrémités d'une arête sont très connectés ce qui implique que l'arête est importante. Par contre, si les voisinages sont très peu connectés cela signifie que les deux extrémités sont très probablement dans deux communautés disjointes donc le lien sera faible. Par conséquent, cette métrique a été définie pour que les arrêtes ayant une valeur forte sont à l'intérieur des communautés alors que les arêtes ayant une valeur faible relient les communautés.

Le premier qui s'est intéressé à cette mesure est Jourdan [4], il l'appela force d'une arête. Néanmoins, la méthode proposée ne peut être appliqué dans notre cas du fait que la méthode proposée dans [4] ne s'applique qu'aux cas des graphes non pondérés. Ainsi, nous allons développer notre approche à partir de ses travaux.

3.2 La méthode

Pour faire le partitionnement dans un graphe $G = (V, E, W)$ basé sur l'importance des arêtes, nous allons diviser les voisins de *u* et *v* en trois ensembles distincts (voir figure2.3) :

- *L* (*u*) : L'ensemble des voisins de *u* qui ne sont pas voisins de *v*.
- *L* (*v*) : L'ensemble des voisins de *v* qui ne sont pas voisins de *u*.
- *L (u,v)* : L'ensemble des voisins communs à *u* et *v*.

L'importance d'une arête est une mesure en fonction de liens entre $L(u,v)$, $L(v)$ et les liens entre les ensembles $L(u,v)$, $L(u)$ de même entre les ensembles $L(v)$, $L(u)$ et les liens dans l'ensemble $L(u,v)$ ce qui permet de mesurer tous les liens possible entre $N(u)-\{v\}$ et $N(v)-\{u\}$ et donc mesurer la connectivité des voisinages de l'arête *uv*.

Soit *r* la fonction définie comme suit : Si *A* et *B* sont deux sous ensembles de *V*:

$$r(A,B) = \sum_{i \in A,\ j \in B,\ ij \in E} w_{ij}$$

La fonction *r* représente la somme des poids de toutes les arêtes entre les deux sous ensembles *A* et *B* de *V*.

Et *s* la fonction définie par :

$$s(A) = \sum_{ij \in E(A)} w_{ij}$$

Où *A* est un ensemble de l'ensemble *V*.

La fonction *s* représente la somme des poids de toutes les arêtes dans le sous ensembles *A* de *V*.

A) Définition de la mesure de l'importance d'une arête

On définit la mesure d'importance d'une arête *uv* par:

$$IP(uv) = \frac{r(L(u),L(v)) + r(L(u),L(u,v)) + r(L(v),L(u,v)) + s(L(u,v))}{|L(u)|\,|L(v)| + |L(u)|\,|L(u,v)| + |L(v)|\,|L(u,v)| + \frac{|L(u,v)|\,(|L(u,v)-1|)}{2}}$$

Remarque bien que $0 \le IP(uv) \le 1$.

B) Identification des communautés

La notion d'importance d'une arête a été définie pour que les arêtes ayant une valeur faible relient les communautés et que les arêtes qui sont à l'intérieur de ces communautés ont une valeur élevée. Mais la notion de faible ou de forte importance d'une arête est relative à un seuil noté α.

Alors, pour une arête *e*, on dit qu'elle est de forte importance (respectivement de faible importance) si $IP(e) \ge \alpha$ (respectivement $IP(e) \prec \alpha$).

Ainsi, pour déterminer les communautés dans un graphe d'association de termes à partir de la notion de l'importance d'une arête, on supprime toutes les arêtes qui ont une importance inférieure au seuil α (i.e. les arêtes de faible importance). Les communautés seront les composantes connexes engendrées par les arêtes sélectionnées.

S'il existe des sommets non classés (i.e. des sommets isolés dans le graphe construit par les arêtes sélectionnées), alors, il paraît plus probable d'affecter ces sommets vers d'autres clusters que de constituer des clusters à un seul sommet [2].

Algorithme: PARTITIONNEMENT

Données: *Un graphe d'association de termes* $G = (V, E, W)$*, un seuil* α.

Résultat: *Une partitionnement* C_α *du graphe* $G = (V, E, W)$.

Début

(1) Poser $E = \{e_1, e_2, ..., e_m\}$

(2) Pour chaque arête e dans E, on calcule la mesure d'importance $IP(e)$

(4) On détermine les ensembles $H_\alpha = \{e \in E / IP(e) \geq \alpha\}$.

(5) $G_\alpha = (V, H_\alpha)$ *un graphe partiel de G.*

(6) Soit $C_\alpha = (c_1^{\alpha}, c_2^{\alpha}, \ldots, c_p^{\alpha})$ *où* c_i^{α} *est une composante connexe dans le graphe* G_α.

(7) Poser $I = \{i \in \{1,..., p\} / |c_i^{\alpha}| = 1\}$ *et* $J = \{i \in \{1,..., p\} / |c_i^{\alpha}| \neq 1\}$

(8) poser $K = \bigcup_{i \in I} c_i^{\alpha}$

(9) Trouver l'ensemble $L = E(\bigcup_{i \in I} c_i^{\alpha}, \bigcup_{i \in J} c_i^{\alpha})$.

(10) Classer les éléments de L selon l'ordre décroissant de l'importance d'arête s'il existe plusieurs arêtes qui ont même importance alors on les classe selon l'ordre décroissant de leurs indices.

(11) Pour chaque arête uv $(u \in \bigcup_{i \in I} c_i^{\alpha}, v \in c_s^{\alpha}$ *avec* $s \in J)$ *de L pris dans l'ordre*

décroissant d'importance d'arête.

(i) Affecter u à la classe c_s^{α}.

(ii) Eléminer toutes les arêtes de L qui ont u comme sommet incident. .

(11) Si $\bigcup_{i \in J} c_i^{\alpha} = V$ *. Terminer, on obtient un partitionnement* $(c_i^{\alpha})_{i \in J}$ *relatif au seuil* α.

(12) Sinon, on revient à (4).

Fin

Pour un seuil donné, on trouve qu'un seul partitionnement par l'algorithme PARTITIONNEMENT

3.3 L'algorithme

Le principe de l'algorithme de partitionnement d'un graphe d'association de termes par l'approche d'importance d'une arête se base sur le choix d'un seuil qui maximise la mesure de qualité à partir de l'algorithme PARTITIONNEMENT.
Le choix du seuil par cette approche sera itératif, c'est à dire que nous devons appliquer le processus en calculant une série de partitionnement correspondant à des différentes valeurs du seuil. La valeur choisie sera celle qui correspond à la meilleure mesure de qualité ψ du clustering [2], c'est à dire un clustering qui maximise la valeur de cette mesure avec :

$$\psi(C)=\frac{\sum_{i=1}^{i=p}\sum_{vv'\in E(c_i)} w_{vv'}}{\sum_{i}\frac{|V(c_i)|\times(|V(c_i)|-1)}{2}}-\frac{\sum_{i,j\ i\prec j}^{p}\sum_{v\in c_i, v'\in c_j, vv'\in E} w_{vv'}}{\sum_{i,j\ i\prec j}^{p}|V(c_i)|\times|V(c_j)|}$$

Où $C=(c_1,c_2,\ldots,c_p)$ un partitionnement du graphe *G*.

La mesure de qualité ψ du clustering $C=(c_1,c_2,\ldots,c_p)$ minimise les poids d'arêtes entre les clusters (poids d'arêtes inter-clusters) et maximise les poids d'arêtes à l'intérieur des clusters (poids d'arêtes intra-clusters).

Maintenant on peut écrire l'algorithme pour trouver un partitionnement d'un graphe d'association de termes à partir de la méthode de l'importance d'une arête.

Algorithme: Méthode de partitionnement par la mesure d'importance d'arête.

Données: *Un graphe d'association de termes* $G=(V,E,W)$.

Résultat: *Un partitionnement C du graphe G.*

Début

(1) Poser $F'=\{IP(e)/\ e\in E\}$

(2) Pour chaque valeur du seuil $\sigma\in F'$.

(i) Trouver un partitionnement C_α *à partir de l'algorithme PARTITIONNEMENT.*

(ii) Calculer la mesure de qualité $\psi(C_\alpha)$ *du partitionnement* C_α.

(3)Trouver un seuil $\sigma'\in F'$ *tel* $\psi(C_{\sigma'})=Max\{\psi(C_\sigma)/\ \sigma\in F'\}$.

(4) $C_{\sigma'}$ *relatif au seuil* σ' *est le partitionnement basé sur l'approche d'importance d'une arête.*

Fin

---

4 Application

Notre objectif est d'appliquer la méthode précédente sur un corpus afin d'aider l'expert à analyser les résultats.

Le corpus sur lequel nous travaillons a été obtenu en utilisant la base de données PASCAL. La base PASCAL est une base de données bibliographiques multidisciplinaire qui permet à l'utilisateur de trouver des références bibliographiques provenant de domaines connexes non couverts par des bases de données spécialisées. C'est un élément particulièrement appréciable dans le cadre d'une recherche de première approche ou sur un thème transversal. Elle constitue un réservoir de 14 millions de références bibliographiques en sciences, technologie et médecine depuis 1973. PASCAL couvre la littérature internationale tout en donnant une priorité au

traitement de la littérature française et européenne qui représente plus de 40 % des documents.

Nous utiliserons les résultats du traitement d'un corpus de références extraites de la base PASCAL, entre 1987 et 2005, dans le domaine de traitement automatique de langage: « analyse » et « texte ». Ce corpus comprend 2390 références.

4.1 L'extraction des candidats termes

L'extraction de termes à partir de données textuelles est une discipline émergente, construite par l'assemblage de diverses disciplines : « …statistiques, intelligence artificielle, base de données,…», elle peut être définie comme étant la tache qui consiste à identifier de l'information bien précise d'un texte en langue naturelle et à le représenter sous forme structurée dans le but de réduire l'effort intellectuel humain. Elle sert donc à retrouver dans un ou plusieurs documents donnés les termes discriminants et utiles à la compréhension du discours. Elle consiste donc à parcourir les immenses volumes des données textuelles, à la recherche de connaissances.

A partir de ces approches, il existe actuellement un grand nombre de logiciels dits de terminologie qui ont été développées afin d'extraire les termes à partir d'un corpus, parmi ces logiciels spécifiques à la terminologie on a choisi le logiciel INTEX ainsi on a obtenu 1023 termes candidats selon le tableau suivant :

Nombre de termes	1023
Nombre de notices	2390

4.2 Recherche des classes

Le résultat de la classification par la notion d'importance d'une arête est une partition des termes de $\mathcal{T}$ en classes structurées mais disjointes (clusters), même si les clusters peuvent entretenir des relations avec d'autres clusters. Ainsi on a obtenu 40 clusters Un cluster est composé de :

une liste de mots-clés,
une liste d'associations internes,
une liste d'associations externes, une étiquette,

Le choix d'un terme représentatif pour nommer le cluster est basé sur une heuristique. Nous choisissons le terme de la liste de termes internes qui apparaît le plus grand nombre de fois dans les associations internes et externes.

5 Conclusion et perspectives

Dans cette contribution, une méthode infométrique pour la détermination de classes dans un corpus textuel est proposée. On a proposé une notion d'importance d'une arête qui permet de faire cet partitionnement. Ainsi, une application de cette méthode sur les notices bibliographiques extraient à partir de la base de données bibliographiques PASCAL.

Comme perspective, on propose de faire une station d'analyse de données bibliographiques qui se base sur cette méthode pour le module de infométrique afin de faire une classification thématique de données bibliographiques.

References

1. S. C. Bradford. Sources of information on specific subjects . Engineering, vol 137 (1934) 85-86
2. Harik, H. Eléments de la théorie des graphes et applications dans le domaine de l'information. Thèse de Magister en recherche opérationnelle, USTHB (2006)
3. A.-K. Jain, R.-C. Dubes. Algorithms for Clustering Data. Prentice Hall Advanced Reference Series (1988)
4. Fabien. Jourdan. Visualisation d'information : dessin, indices structuraux et navigation. Applications aux réseaux biologiques et aux réseaux sociaux. Thèse de Doctorat en Informatique, Université de Montpellier II (2004)
5. A. J. Lotka. "The frequency distribution of scientific productivity". Journal of the washington academy of sciences, vol 16, N° 12, p 317-323 (1926)
6. Manolis M. Tsangaris and Jeffrey F. Naughton. A stochastic approach for clustering in object bases. In Proc. of the 1991 ACM SIGMOD International Conference on Management of Data, pages 12–21, Denver, CO, (USA). ACM Press (1991)
7. Manolis M. Tsangaris and Jeffrey F. Naughton. On the performance of object clustering techniques. In Proc. of the 1992 ACM SIGMOD International Conference on Management of Data, pages 144–153, San Diego, CA, (USA). ACM Press (1992)
8. Zipf, G, K. Human behaviour and the principale of least effort: An introduction to Human ecology Reading. Mass: Adisson Wesley (1949)

Towards easy prototyping of pattern mining problems

Frédéric Flouvat[1], Fabien De Marchi[2], and Jean-Marc Petit[1]

[1] LIRIS, INSA-Lyon, F-69621, France
frederic.flouvat@liris.cnrs.fr
jean-marc.petit@liris.cnrs.fr
[2] LIRIS, Université Lyon I, F-69621, France
fabien.demarchi@liris.cnrs.fr

Abstract. In the last decade, plenty of algorithms, benchmarks, and experimental studies have been carried out for pattern mining problems. In this paper, we focus on the special class of pattern mining problems known to be "representable as sets". In this setting, the main contribution of this paper is to take advantage of the common theoretical background of these problems from an implementation point of view by proposing a library of efficient data structures and algorithms for pattern mining. Thus, every problem fulfilling the theoritical requirements could be implemented with only minimal effort. According to our first results, the programs obtained using our library offer a very good tradeoff between performances and simplicity of their development.

Keywords : pattern mining, inductive databases, boolean lattices, C++ library.

1 Introduction

In the last decade, plenty of algorithms, benchmarks, and experimental studies have been carried out for pattern mining problems. Pattern mining problems encompass the discovery of patterns such as itemsets, sequences, trees, graphs, and many other structures. Varied approaches to these problems appear in numerous papers, e.g. [1–4].

Given the proliferation of pattern mining implementations, there is a pressing need to understand where the key issues are. As shown in FIMI and OSDM workshops in 2003, 2004 and 2005 [5–7], best known implementations use quite sophisticated low level code to optimize for instance basic IO routines, which raise interesting issues while being far away from trying to understand why and under what conditions one algorithm would outperform another.
Morevover, it was advocated that algorithms for frequent itemset mining can be used as a building block for other, more sophisticated data mining problems. This is especially true for pattern mining problems known to be "representable as set" [8], as for instance, frequent itemset mining and variants [1, 9, 10], functional

or inclusion dependency inference [11–13] or learning monotone boolean function [14]. Roughly speaking, the class of problems we are interested in and referred to as "isomorphic to boolean lattices" in this paper, consists of pattern mining problems for which: 1) the predicate defining the interestingness criteria has to be monotone with respect to the partial order of patterns, 2) there exists a bijective function f between the set of patterns and some finite set, 3) the partial order among patterns is preserved, i.e. $X \preceq Y \Leftrightarrow f(X) \subseteq f(Y)$.

In this setting, the basic idea is to say that algorithms devised so far should be useful to answer these tasks and available open source implementations a great source of "know how". Unfortunately, it seems rather optimistic to envision the application of most of publicly available implementations of frequent itemset mining algorithms, even for closely related problems.
Indeed, even if the development of efficient data structures for managing huge sets of set is definitely useful, loading the database in main memory using sophisticated data structure specially devised for a monotone predicate turns out to give very efficient algorithms but deserve other data mining tasks. As a consequence, low level implementations hamper the rapid advances in the field.

In this paper, we focus on the special class of pattern mining problems known to be "representable as sets" [8]. Our objective is twofold:

- to make easier the development of data mining implementations by offering in a black box state of the art data structures and algorithms for free.
- to guide the developers in the key stages of the design of his/her implementation.

Thus, according to the problem characteristics, she/he will have to focus only on the core strategy to be used to implement a scalable implementation, without worrying about technical details useless for prototyping.

Paper contribution In this setting, the main contribution of this paper is to take advantage of the common theoretical background of these problems from an implementation point of view by providing efficient data structures for boolean lattice representation and several implementation of well known algorithms such as levelwise algorithm and dualization-based algorithm. By the way, every problem fulfilling these requirements could be implemented with only minimal effort, i.e. the programmers do not have to be aware of low level code, customized data structures and algorithms being available for free.
A library, called *iZi*[3], has been devised and applied to several problems such as itemset mining or constraint mining in relational database. According to our first results, the programs obtained using the library have very interesting performances in comparison with the simplicity of their development. Some methodological guidelines are also provided to guide the programmers both at the theoretical level and at the C++ code level. The library is freely available on the Web [15].

[3] derived from the phonetic of the word "easy": ['iːzɪ]

Paper organization In section 2, we introduce the theoretical framework used (subsection 2.1) and guidelines to use it (subsection 2.2). This section also discusses how state of the art solutions can be used in our generic context (subsection 2.3). The section 3 presents our contribution: the *iZi* library. The architecture of the library is given in subsection 3.1 and is followed by a quick start guide (subsection 3.2). Experimentations are described in section 4. Section 5 discusses the interest of our proposition w.r.t. to related works. The last section concludes and gives some perspectives to this work.

2 Framework

2.1 Theoritical components

We recall in this section the theoretical KDD framework defined in [8] for interesting patterns discovery problems.

Given a database d, a finite language $\mathcal{L}$ for expressing patterns or defining subgroups of the data, and a predicate Q for evaluating whether a pattern $\varphi \in \mathcal{L}$ is true or "interesting" in d, the discovery task is to find the theory of d with respect to $\mathcal{L}$ and Q, i.e. the set $Th(\mathcal{L}, d, Q) = \{\varphi \in \mathcal{L} \mid Q(d, \varphi) \text{ is true}\}$.

Let us suppose a specialization/generalization relation between patterns of $\mathcal{L}$. Such a relation is a partial order $\preceq$ on the patterns of $\mathcal{L}$. We say that φ is more general (resp. more specific) than θ, if $\varphi \preceq \theta$ (resp. $\theta \preceq \varphi$).

The predicate Q is said to be monotone (resp. anti-monotone) with respect to $\preceq$ if for all $\theta, \varphi \in \mathcal{L}$ such that $\varphi \preceq \theta$, if $Q(d, \varphi)$ is true (resp. false) then $Q(d, \theta)$ is true (resp. false). As a consequence, if the predicate is monotone (resp. anti-monotone), the set $Th(\mathcal{L}, d, Q)$ is upward (resp. downward) closed, and can be represented by any of the following set:

- its *positive border*, denoted by $\mathcal{B}d^+(\ Th(\mathcal{L}, d, Q)\ )$, made up of the most specialized TRUE patterns when $Th(\mathcal{L}, d, Q)$ is downward closed, and the most generalized TRUE patterns when $Th(\mathcal{L}, d, Q)$ is upward closed;
- its *negative border*, denoted by $\mathcal{B}d^-(\ Th(\mathcal{L}, d, Q)\ )$, made up of the most generalized FALSE patterns when $Th(\mathcal{L}, d, Q)$ is downward closed, and the most specialized FALSE patterns when $Th(\mathcal{L}, d, Q)$ is upward closed.

The union of this two borders is called the *border of* $Th(\mathcal{L}, d, Q)$, and is denoted by $\mathcal{B}d(\ Th(\mathcal{L}, d, Q)\ )$. The search space can be respresented by a lattice (see figure 1) composed of all the patterns of the language $\mathcal{L}$ and ordered w.r.t. the partial order $\preceq$. Then, the border caracterizes this search space. If the predicate is monotone (resp. anti-monotone), all the patterns "above" this border are true (resp. false), and all the patterns "below" are false (resp. true).

Note that when an interestingness measure is associated to patterns of $Th(\mathcal{L}, d, Q)$, the borders may not be sufficient for recovering all TRUE patterns together with their value. It is the case for the frequency measure defined for itemsets in a transaction database.

Last hypothesis of this framework, the problem must be representable as sets via an isomorphism, i.e. the search space can be represented by a boolean lattice.

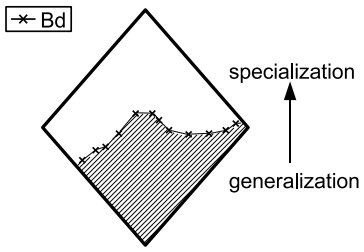


Fig. 1. Border and search space

Let $(\mathcal{L}, \preceq)$ be the ordered set of all the patterns defined by the langage $\mathcal{L}$. Let I be a finite set of items. The problem is said to be *representable as sets* if an isomorphism exists from $(\mathcal{L}, \preceq)$ to $(2^I, \subseteq)$, such that:

$$X \preceq Y \Longleftrightarrow f(X) \subseteq f(Y)$$

2.2 Guidelines

Once a problem fits in this framework, a program can be rapidly developped using our proposal. However, it is not always trivial to ensure this fitting; from our experience [16, 11, 17, 18], we give some methodological insights in this section, illustrated through an example which is key discovery in databases.

What kind of problems ? Problems have to be *enumeration problem under constraints*, i.e. of the form "enumerate all the patterns that verify a condition". When the condition must be verified in a data set, the term "enumerate" is commonly replaced by "extract". Frequently, the problem specification precises that patterns must be maximal or minimal: maybe only the positive of negative border is wondered.

Langage, predicate and monotony Once a problem is suspected to fit into the framework, its components must be properly defined to go further.

1. Precising the patterns: what is the search space ? it implies to define a langage or a formal characterization of what patterns are mining for.
2. Pointing out the predicate with a mathematical sentence as: A pattern X is true if $C(d, X)$, where C is a condition expressed over data.

These specifications are helpful to demonstrate the monotonicity of the predicate. The question is: if a pattern X is true (or false), does another more specific (i.e. less simple) pattern always true (resp.false) ? Note that if the problem is to search for maximal or minimal patterns verifying some properties, then the predicate monotonicity is almost sure. Given the sens of the order, it might be monotone or anti-monotone.

Isomorphism to boolean lattice Sometimes, the isomorphism is trivial, as it is the case for frequent itemsets problems: the set of candidates is naturally expressed as a powerset. In other cases, a more deep study must be performed. An important remark is that a necessary condition is that the number of candidates, when executing the program, must

As an example, we consider the problem of discovering the keys of a relation [19].

Example 1:
What kind of problems ?
Let us consider the key discovery problem in a relation, which can be enounced as follows: Let r be a relation over a schema R, extract the (minimal) keys satisfied in r.
Langage, predicate and monotony
The patterns are : $\{X \mid X \subseteq R\} = \mathcal{P}(R)$. X is true if X is a superkey, i.e. if $|\pi_X(r)| = |r|$, where $\pi_X(r)$ is the projection onto X over r. It is clear that any superset of a superkey is also a superkey, it justifyes that only minimal keys are really interested. One can deduce that minimal keys constitute the positive border of superkeys, with natural set inclusion.
Isomorphism to boolean lattice
The transformation function here is the identity since patterns are sets.

2.3 Generic Algorithms and data structures

Even if this framework has been frequently used at a theoretical level, it has never been exploited at a technical point of view. The classical way used to solve pattern mining problems is to develop had-hoc solutions, with very specialized data structures and optimization technics. If such a solution leads to efficient programs in general, it requires a huge amount of work to obtain a sound and operational tool. Moreover, if problem specifications slightly differ over time, a consequent effort will be made to identify what parts of the program must be updated.

One of our goal is to factorize some technical solutions which can be common to any pattern mining problem representable as sets. We are interested in algorithms and data structures that apply directly on sets, since they can be applied without any change for any problem, exploiting the isomorphic transformation.

Our solution reuse some previous works done for frequent itemset mining (FIM), which is a problem "directly" representable as sets, i.e. the isomorphic transformation is the identity.

Algorithms Currently, our library implements two algorithms from the multitude that has been proposed for the FIM problem. The first is the well known *Apriori* algorithm [20], which is still in course for discovering the whole theory for *not too large positive borders* [21]. The second one is *ABS* [22], well adapted to discover any positive border when the negative border is not too large. We argue these choices with the following consideration: they are really generic in our framework,

since they only rely on set operations and predicate evaluations. They do not exploit any specific property of the predicate, as it is the case for FP-growth like algorithms [23] or contributions using condensed representations [24].

Note that other solutions like a pure depth-first approach are also generic and potentially interested for some problems. We plan to offer such a strategy in future release of the library.

Data structures Since the generic part of our library only manipulates sets, we use a data structure based on prefix-tree specially devoted to this purpose [25, 26]. They have not only a power of compression by factorizing common prefix in a set collection, but are also well adapted for inclusion and intersection tests, which are basic operations when considering sets. Of course, as for algorithms, one can imagine to extend our library with alternative structures for sets, like bitmaps.

3 A generic library for patterns discovery

Based on the theoretical framework presented in section 2, we propose a C++ library, called *iZi*, for this family of problems. The idea which guides this work is to offer a toolbox for a rapid development of an efficient and robust program, using a selected algorithm. Note that the development and structure of the proposed library result from our experience when implementing and adapting algorithms to solve these problems.

3.1 Architecture

The figure 2 represents the architecture and the "workflow" of our library.

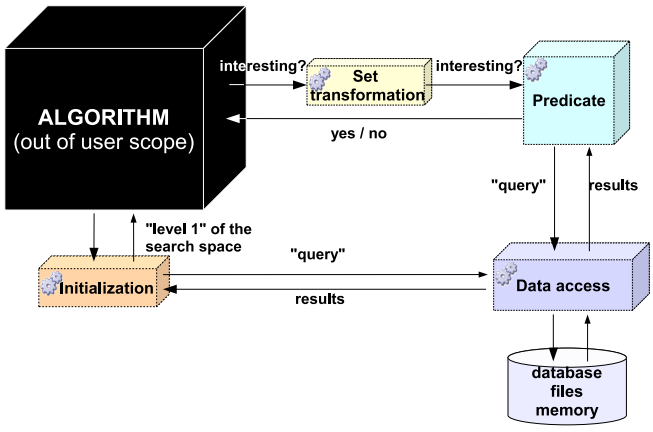


Fig. 2. *iZi* architecture

As shown by the figure 2, the library is composed of 5 types of components: *algorithm*, *set transformation* function, *predicate* function, *data access* component and *initialization* function.

The *algorithm* component represents generic algorithms implementations provided in the library and used to solve users pattern mining problems. As shown by the figure 2, algorithms are decoupled of the problems and are black box for the users. Each algorithm can be used directly to solve any problem fitting in the studied framework without modifications. This leads to the rapid construction of robust programs for the user without having to implement complex algorithms. The other components are specifics to a given problem. Consequently, to solve a new problem, users may have to implement some of these components. However, a user can also reuse existing components developed for other problems.

Another important aspect of our library is that data access is totally decoupled of all the other components (see figure 2). Currently, data access in most of the implementations is tightly coupled with the algorithm implementation and the problem which limits their use.

The functioning of the library is presented in the figure 2. First, the *algorithm* is initialized (*initialization* function) with the patterns corresponding to singletons in the set representation, eventually by accessing data (*data access* component). Then, during the execution of the algorithm, the predicate is used to test each pattern using the *data*. Recall that as defined in the framework, algorithms of the library only manipulate set elements. Consequently, before testing an element, the algorithm may have to use a *set transformation* function to transform each set generated into the corresponding pattern.

3.2 Getting started with *iZi*

To install *iZi*, the user only has to download the files from our website [15] and to include the header files in her/his source code. Then, she/he can directly use the library.

Before implementing any components, the user has to verify that her/his problem is not already provided by the library. If the problem is already implemented, the *Predicate* and *Initialization* components (figure 2) can be used without modifications, otherwise they must be redeveloped. However, if the problem is close to a problem provided with the library, the related components can be used as examples or derived to process the new problem. This alternative has in particular been tested when implementing frequent itemsets mining and frequent essential itemsets mining problems.

In the same way, the library provides data access components to the user for implementing her/his problem, thus allowing to directly access several data types and data sources.

The table 1 shows the differents problems, data types and sources provided with *iZi*. The library already contains 4 problems, and 4 data access types and sources. For itemset mining in transactional databases, the format considered is the FIMI file format which was defined for the FIMI workshops [5, 6], and is

Data type	Problem	File (format)	DBMS
tabular	inclusion dependencies	CVS FDEP	MySQL
tabular	keys	CVS FDEP	MySQL
binary	frequent itemsets	FIMI	
binary	frequent essential itemsets	FIMI	

Table 1. Problems, data types and formats provided with *iZi*

the most comonly used data format for this family of problems. For constraint mining in relational databases, components were also developed to access data in files (CSV format of *Excel* and *FDEP* format [27]) and in the *MySQL* DBMS.

Once the problem and data access components are defined, the analyst must choose a strategy to explore the search space. Currently, the library provides 4 search space strategies:

- two bottom-up algorithms (figure 3): an *Apriori*-like [20, 8] and *ABS*.
- two top-down algorithms (figure 4): an inverted version of *Apriori* and inverted version of *ABS*.

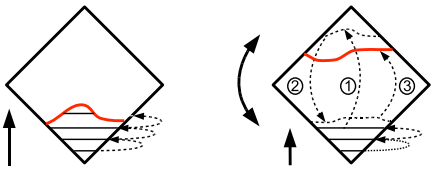


Fig. 3. The bottom-up algorithms: *Apriori* (left) and *ABS* (right)

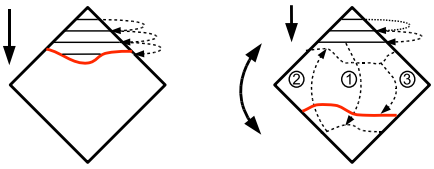


Fig. 4. The top-down algorithms: inverted *Apriori* (left) and inverted *ABS* (right)

First, the user has to identify algorithms "compatibles" with her/his predicate properties and the solution she/he wants to discover. Note that from a theoretical point of view, the positive (resp. negative) border of a predicate is the negative (resp. positive) border of the inverted predicate, i.e. $\mathcal{B}d^+(Th(\mathcal{L}, d, Q)) = \mathcal{B}d^-(Th(\mathcal{L}, d, \neg Q))$ and $\mathcal{B}d^-(Th(\mathcal{L}, d, Q)) = \mathcal{B}d^+(Th(\mathcal{L}, d, \neg Q))$. Consequently by inverting a monotone (resp. anti-monotone) predicate Q, bottom-up (resp. top-down) approaches could be used to mine the borders of this predicate. For this reason, the library provides a generic component to invert predicates.

The table 2 presents all these possibilities.

Secondly, the analyst may have to choose an algorithm among all the algorithms "compatibles" with her/his problem specificity. To do this choice, the analyst could use her/his own knowledge about the domain or the data.
This is typically the case when choosing between a bottom-up or a top-down

	monotone predicate Q	anti-monotone predicate Q	inverted monotone predicate $\neg Q$	inverted anti-monotone predicate $\neg Q$
Th	inverted *Apriori*	*Apriori*		
$\mathcal{B}d^+$	inverted *Apriori* inverted *ABS*	*Apriori* *ABS*	*Apriori* *ABS*	inverted *Apriori* inverted *ABS*
$\mathcal{B}d^-$	inverted *Apriori* inverted *ABS*	*ABS* *ABS*	*Apriori* *ABS*	inverted *ABS* inverted *ABS*

Table 2. Algorithms w.r.t. solution and predicate

strategy. As shown by the figure 5, the efficiency of each strategy is influenced by the "position" of the solutions in the search space.

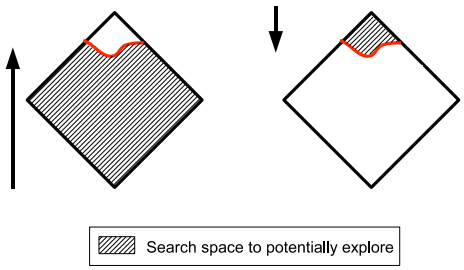


Fig. 5. Search space explored by bottom-up and top-down strategies w.r.t. solutions

The choice between an *Apriori*-like approach or an *ABS*-like approach could also be done based on expert knowledge. Actually, as shown in [21, 22], *Apriori* is more particularly efficient when interesting patterns are "low" in the search space, and *ABS* when pattens of the negative border are "far" from patterns of the positive border.

4 Experimentations

Implementation of a new problem It is a very hard task to evaluate the difficulty for a user to develop a program with our library.
From our past experience in the development of pattern mining algorithms [22, 18, 28], we noted that the adaptation of existing implementations was extremely difficult. In some cases, it implies the redevelopment of most of the implementation and could take more time than developing a new program from scratch.
As shown in the previous section, four problems have been implemented in our library to test the difficulty of implementing a new predicate or changing the data access. As indication, the use of our library to implement a program for the key mining problem was done in less than one working day.

Performance In this section, we present some experimental results of our library for frequent itemset mining. This problem was more particularly choosen for our experiments because there is lot of ressources (algorithms implementations, datasets, benchmarks...) freely available on internet [29]. Whereas, for other problems such as key mining, even if algorithms implementations are sometimes available, it is difficult to have access to the datasets. As an example, initially we plan to compare *iZi* with the algorithm proposed in [19] for key mining. Unfortunately, the implementation was not available and the authors could not provide us their datasets for confidentiality reasons.

The discovery of frequent itemsets is a problem where implementations are very optimized, specialized, and consequently very competitives. The most performant ones are often the results of many years of research and development. In this context, the objective of these experimentations is to show that a generic library such as *iZi* could have comparable performances with specialized implementations.

Experiments have been done on some FIMI databases [5, 6]. We compared our *Apriori* generic implementation together with two others devoted implementations: one by B. Goethals [30] and one by C. Borgelt [31]. The first one is a quite natural version, while the second one in, to our knowledge, is the best existing *Apriori* implementation, developed in *C* and strongly optimized.

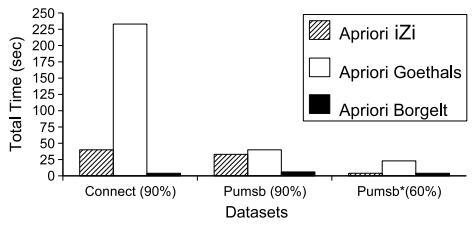


Fig. 6. Comparison of three *Apriori* implementations

The figure 6 shows execution times for some supports of databases *Connect*, *Pumsb* and *Pumsb∗*. One can observe that the performance of our generic version are between the two other implementations. These results are very encouraging, in regards of the simplicity to obtain an operational program.

5 Related works

Most of the algorithm implementations for pattern mining appear as independent and specialized programs, developed mainly to evaluate the effectiveness of the algorithm for a given problem. It is for example the case of all the implementations made available by FIMI workshops [5, 6] for frequent itemsets mining. Other works propose free data mining and machine learning packages.

As an example, the *Illimine* project [32] consider mining data cubes, association rules, sequential pattern, graphs and clustering. In this project, tools have been developed in order to facilitate the use of programs.

However, all these works are dedicated and optimized for the resolution of a specific problem. Thus, these programs cannot be directly used to solve other problems even if they are equivalent. Moreover, the source codes are not always available, and when they are, their modification is very difficult and sometimes impossible due to the complexity of the implementation.

To our knowledge, only one work intents to simplify the programmer task: the DMTL library (Data Mining Template Library) [33].

The DMTL library DMTL is a C++ library composed of algorithms and data structures for frequent pattern mining. This library allows the user to process many types of frequent patterns (sets, sequences, trees and graphs) using generic algorithms implementations. Actually, DMTL could be used to mine any types of patterns respresentable as graphs. Moreover, the data is decoupled of the algorithms, and could be stored in memory, files, Gigabase databases (an embedded object relational database), and PSTL components (a library of persistent containers). Nevertheless, surprisingly DMTL cannot process the FIMI format which is the most commonly used for frequent itemsets mining. This library currently contains two search space strategies: a levelwise *Apriori*-like [20, 8] and a depth-first approach. These algorithms could be used to mine all the frequent patterns of a given database.

However, to our point of view, the DMTL library has three main disavantages. First, this library cannot be used to solve other problems that frequent pattern mining, even other closely related problems such as mining frequent free itemsets. Secondly, only the enumeration of all the frequent patterns is possible. DMTL cannot discover the positive or negative borders of frequent patterns. Thirdly, only "classic" algorithms are implemented, and these algorithms test lot of patterns. None of the strategies use for example patterns of the positive border to prune the search space.
Note that our library intents to give a solution to these limits by allowing the resolution of any pattern mining problem fitting in the framework, the discovery of the theory and the borders, and the use of non classical and efficient algorithms such as *ABS*.

6 Discussion and perspectives

In this paper, we consider the discovery of interesting patterns from data, *when problems are representable as sets* [8]. This work concerns many data mining problems with a new point of view: the rapid development of an efficient and robust program, using a selected algorithm. From a theoretical framework, we propose a C++ library, called *iZi*, which aims at facilitate the development of a scaling solution, using generic algorithms and data structures. Algorithms details and optimizations are transparent for user, and only some specific aspects

of his/her problem must be coded. Moreover, data access is totally decoupled of the other components which enables to use the same code for many data sources wihtout modifications. The experimental results done are encouraging, in comparison with the rapidity and simplicity for devising an operational program.

There are numerous perspectives to this work. First, on a theoretical level, we plan to study some constraint relaxation for accepting more problems, like frequent sequence or tree mining. We also plan to develop a declarative version for such mining problem, with a query language and a cost model, allowing an optimizer to automatically chose the search strategy to liberate the user of this difficult task.

References

1. Agrawal, R., Imielinski, T., Swami, A.N.: Mining association rules between sets of items in large databases. In Buneman, P., Jajodia, S., eds.: SIGMOD Conference, ACM Press (1993) 207–216
2. Zaki, M.J.: Scalable algorithms for association mining. IEEE Trans. Knowl. Data Eng. **12**(2) (2000) 372–390
3. Termier, A., Rousset, M.C., Sebag, M.: Treefinder: a first step towards xml data mining. [34] 450–457
4. Asai, T., Arimura, H., Abe, K., Kawasoe, S., Arikawa, S.: Online algorithms for mining semi-structured data stream. [34] 27–34
5. Bayardo Jr., R.J., Zaki, M.J., eds.: FIMI '03, Proceedings of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations, Melbourne, Florida, USA, November 19, 2003. In Bayardo Jr., R.J., Zaki, M.J., eds.: FIMI. Volume 90 of CEUR Workshop Proceedings., CEUR-WS.org (2003)
6. Bayardo Jr., R.J., Goethals, B., Zaki, M.J., eds.: FIMI '04, Proceedings of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations, Brighton, UK, November 1, 2004. In Bayardo Jr., R.J., Goethals, B., Zaki, M.J., eds.: FIMI. Volume 126 of CEUR Workshop Proceedings., CEUR-WS.org (2004)
7. Goethals, B., Nijssen, S., Zaki, M.J.: Open source data mining: Workshop report. In: OSDM'05, Proceeding of the ACM SIGKDD Open Source Data Mining Workshop, Chicago,USA. (2005)
8. Mannila, H., Toivonen, H.: Levelwise search and borders of theories in knowledge discovery. Data Min. Knowl. Discov. **1**(3) (1997) 241–258
9. Bastide, Y., Taouil, R., Pasquier, N., Stumme, G., Lakhal, L.: Mining frequent patterns with counting inference. SIGKDD Explorations **2**(2) (2000) 66–75
10. Boulicaut, J.F., Bykowski, A., Rigotti, C.: Free-sets: A condensed representation of boolean data for the approximation of frequency queries. Data Min. Knowl. Discov. **7**(1) (2003) 5–22
11. Lopes, S., Petit, J.M., Toumani, F.: Discovering interesting inclusion dependencies: application to logical database tuning. Inf. Syst. **27**(1) (2002) 1–19
12. De Marchi, F., Petit, J.M.: Zigzag: a new algorithm for mining large inclusion dependencies in database. In: ICDM, IEEE Computer Society (2003) 27–34
13. Koeller, A., Rundensteiner, E.A.: Heuristic strategies for inclusion dependency discovery. In Meersman, R., Tari, Z., eds.: CoopIS/DOA/ODBASE (2). Volume 3291 of Lecture Notes in Computer Science., Springer (2004) 891–908
14. Angluin, D.: Queries and concept learning. Machine Learning **2**(4) (1987) 319–342

15. Flouvat, F., De Marchi, F., Petit, J.M.: The iZi library website, http://liris.cnrs.fr/ffflouvat/librairie/index.html
16. Lopes, S., Petit, J.M., Lakhal, L.: Efficient discovery of functional dependencies and armstrong relations. In Zaniolo, C., Lockemann, P.C., Scholl, M.H., Grust, T., eds.: EDBT. Volume 1777 of Lecture Notes in Computer Science., Springer (2000) 350–364
17. Marchi, F.D., Lopes, S., Petit, J.M., Toumani, F.: Analysis of existing databases at the logical level: the dba companion project. SIGMOD Record **32**(1) (2003) 47–52
18. De Marchi, F., Flouvat, F., Petit, J.M.: Adaptive strategies for mining the positive border of interesting patterns: Application to inclusion dependencies in databases. In Boulicaut, J.F., De Raedt, L., Mannila, H., eds.: Constraint-Based Mining and Inductive Databases. Volume 3848 of Lecture Notes in Computer Science., Springer (2005) 81–101
19. Sismanis, Y., Brown, P., Haas, P.J., Reinwald, B.: Gordian: Efficient and scalable discovery of composite keys. In Dayal, U., Whang, K.Y., Lomet, D.B., Alonso, G., Lohman, G.M., Kersten, M.L., Cha, S.K., Kim, Y.K., eds.: VLDB, ACM (2006) 691–702
20. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In Bocca, J.B., Jarke, M., Zaniolo, C., eds.: VLDB, Morgan Kaufmann (1994) 487–499
21. Goethals, B., Zaki, M.J.: Advances in frequent itemset mining implementations: report on fimi'03. SIGKDD Explorations **6**(1) (2004) 109–117
22. Flouvat, F., De Marchi, F., Petit, J.M.: ABS: Adaptive Borders Search of frequent itemsets. [6]
23. Han, J., Pei, J., Yin, Y.: Mining frequent patterns without candidate generation. In Chen, W., Naughton, J.F., Bernstein, P.A., eds.: SIGMOD Conference, ACM (2000) 1–12
24. Uno, T., Asai, T., Uchida, Y., Arimura, H.: Lcm: An efficient algorithm for enumerating frequent closed item sets. [5]
25. Borgelt, C.: Efficient implementations of Apriori and Eclat. [5]
26. Bodon, F.: Surprising results of trie-based fim algorithms. [6]
27. Flach, P.A., Savnik, I.: Database dependency discovery: A machine learning approach. AI Commun. **12**(3) (1999) 139–160
28. Jaudoin, H., Flouvat, F.: Techniques de fouille de données pour la réécriture de requêtes en présence de contraintes de valeurs. In Ritschard, G., Djeraba, C., eds.: EGC. Volume RNTI-E-6 of Revue des Nouvelles Technologies de l'Information., Cépaduès-Éditions (2006) 77–88
29. Goethals, B.: Frequent itemset mining implementations repository, http://fimi.cs.helsinki.fi/
30. Goethals, B.: Apriori implementation. University of Antwerp. http://www.adrem.ua.ac.be/~goethals/
31. Borgelt, C.: Recursion pruning for the apriori algorithm. [6]
32. Han, J., Group, D.M.: IlliMine project. University of Illinois Urbana-Champaign Database and Information Systems Laboratory. http://illimine.cs.uiuc.edu/ (2005)
33. Hasan, M., Chaoji, V., Salem, S., Parimi, N., Zaki, M.: DMTL: A generic data mining template library. In: Workshop on Library-Centric Software Design (LCSD'05), with Object-Oriented Programming, Systems, Languages and Applications (OOPSLA'05) conference, San Diego, California. (2005)
34. Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM 2002), 9-12 December 2002, Maebashi City, Japan, IEEE Computer Society (2002)

A logarithm barrier method for semi-definite programming

Jean-Pierre CROUZEIX * Bachir Merikhi †

December 2005

Abstract

This paper presents a logarithmic barrier method for solving a semi-definite linear program. The descent direction is the classical Newton direction. We propose alternative ways to determine the step-size along the direction which are more efficient than classical line-searches.

1 Introduction

In this paper we present an algorithm for solving the optimization problem:

$$m_d = \inf_y [\, b^t y \, : \, \sum_{i=1}^{m} y_i A_i - C \, \in K, \; y \in \mathbb{R}^m], \qquad (D)$$

where K denotes the cone of $n \times n$ symmetric positive semi-definite matrices, the vector $b \in \mathbb{R}^m$ and the $n \times n$ symmetric matrices C and $A_i, i = 1, .., m$, are given. The dual problem of (D) is :

$$m_p = \max_X [\langle C, X \rangle : X \in K, \; \langle A_i, X \rangle = b_i \; \forall \, i = 1, .., m], \qquad (P)$$

where by $\langle C, X \rangle$ we denote the trace of the matrix $(C^t X)$. It is recalled that $\langle \cdot, \cdot \rangle$ corresponds to an inner product on the space of $n \times n$ matrices.

*LIMOS, Université Blaise Pascal, Campus des Cézaux, 63174 Aubière, France.

†Laboratoire d'optimisation, Université Ferhat Abbas, Algérie, the research of this author has been made possible thanks to a PROFAS grant and the hospitality of Université Blaise Pascal.

These problems are linear. Their feasible sets involving the cone of positive semi-definite matrices, a non polyhedral convex cone, they are called linear semi-definite programs. Such problems are the object of a particular attention since the papers by Alizadeh [1] and [2], as well on a theoretical or an algorithmical aspect, see for instance the following references [1],[2], [3], [4], [6] and [7].

Under suitable conditions, solving (D) is equivalent to solving (P): the optimal solutions of one problem being easily obtained when one optimal solution of the other problem is known. In this paper, the problem (D) is approximated by the problem (D_r), $(r > 0)$,

$$m(r) = \inf\ [\ f_r(y)\ :\ y \in \mathbb{R}^m], \qquad (D_r)$$

where the barrier function $f_r : \mathbb{R}^m \to]-\infty, +\infty]$ is defined by

$$f_r(y) = \begin{cases} b^t y + nr\ln r - r\ln[\det(\sum_{i=1}^m y_i A_i - C)] & \text{if} \quad y \in \widehat{Y}, \\ +\infty & \text{if not.} \end{cases}$$

with

$$\widehat{Y} = \{y \in \mathbb{R}^m : \text{the matrix } \sum_{i=1}^m y_i A_i - C \in \widehat{K}\},$$

and $\widehat{K} = \text{int}(K)$ is the cone of $n \times n$ symmetric positive definite matrices. This problem is solved via a classical Newton descent method. The difficulty is in the line-search: the presence of a determinant in the definition of f_r induces high computational costs in classical exact or approximate line-searches. Here, instead of minimizing f_r along the descent direction d at the current point x, we minimize a function $\widetilde{\theta}$ such that

$$\frac{1}{r}[f_r(x+td) - f_r(x)] = \theta(t) \le \widetilde{\theta}(t) \quad \forall\, t > 0, \quad \theta(0) = \widetilde{\theta}(0), \quad \theta'(0) = \widetilde{\theta}'(0) < 0.$$

This function $\widetilde{\theta}$ needs to be appropriately chosen so that the optimal t is easily obtained and to be close enough to θ in order to give a significant decrease of f_r in the iteration step. We propose in this paper functions θ for which the optimal solution t is explicitly obtained and a good quality of the approximation of θ by $\widetilde{\theta}$ is ensured by the condition $\theta''(0) = \widetilde{\theta}''(0)$.

In the next section, we briefly recall some results in linear semi-definite programming. Section 3 studies the problem (D_r), in particular the behavior of its optimal value and its optimal solutions when $r \to 0$. Section 4 shows how to compute the Newton descent direction. Section 6 is devoted to the determination of efficient approximations $\widetilde{\theta}$, these approximations are deduced from inequalities shown in Section 5. The algorithm is resumed in Section 7 and numerical experiments presented in Section 8 show the efficiency of the approximations when compared with classical line-searches.

2 A brief background in linear semi-definite programming

Throughout the paper, we use the following notation:

$$\begin{array}{ll} Y = \{y \in \mathbb{R}^m : \sum_{i=1}^m y_i A_i - C \in K\}, & F = \{X \in K : \langle A_i, X\rangle = b_i \; \forall\, i\}, \\ \widehat{Y} = \{y \in \mathbb{R}^m : \sum_{i=1}^m y_i A_i - C \in \widehat{K}\}, & \widehat{F} = \{X \in F \,:\, X \in \widehat{K}\}. \end{array} .$$

It is easily seen that $-\infty \leq m_p \leq m_d \leq +\infty$ (weak duality). In this paper we assume that the two following assumptions hold:

(H1) The system of equations $\langle A_i, X\rangle = b_i$, $i = 1, ..., m$ is of rank m.

(H2) The sets $\widehat{Y}$ and $\widehat{F}$ are non empty.

Then it is known that (see for instance [1], [3]):

a) $-\infty < m_p = m_d < +\infty$.

b) The sets of optimal solutions of (P) and (D) are non empty convex compact sets.

c) If $\bar{X}$ is an optimal solution of (P), then $\bar{y}$ is an optimal solution of (D) if and only if

$$\bar{y} \in Y \quad \text{and} \quad (\sum_{i=1}^m \bar{y}_i A_i - C)\bar{X} = 0.$$

d) If $\bar{y}$ is an optimal solution of (D), then $\bar{X}$ is an optimal solution of (P) if and only if

$$\bar{X} \in F \quad \text{and} \quad (\sum_{i=1}^{m} \bar{y}_i A_i - C)\bar{X} = 0.$$

3 Theoretical study of problem (D_r)

Recall that (D_r), $r > 0$, is the problem

$$m(r) = \inf\ [\ f_r(y)\ :\ y \in \mathbb{R}^m], \qquad (D_r)$$

with $f_r : \mathbb{R}^m \to (-\infty, +\infty]$ defined by

$$f_r(y) = \begin{cases} b^t y + nr \ln r - r \ln[\det(\sum_{i=1}^m y_i A_i - C)] & \text{if } y \in \widehat{Y}, \\ +\infty & \text{if not.} \end{cases}$$

We start with the study of this function.

3.1 f_r is a twice differentiable strictly convex function

The following notation will be used in the expressions of the gradient and the Hessian of f_r: given $y \in \widehat{Y}$, we introduce the $m \times m$ symmetric positive definite matrix $B(y)$ and the lower triangular $m \times m$ matrix $L(y)$ such that

$$B(y) = \sum_{i=1}^{m} y_i A_i - C = L(y)L^t(y).$$

Next, for $i, j = 1, 2, \cdots, m$, we define

$$\widehat{A}_i(y) = [L(y)]^{-1} A_i [L^t(y)]^{-1},$$

$$b_i(y) = \text{trace}(\widehat{A}_i(y)) = \text{trace}(A_i B^{-1}(y)),$$

$$\Delta_{ij}(y) = \text{trace}(B^{-1}(y)A_i B^{-1}(y)A_j) = \text{trace}(\widehat{A}_i(y)\widehat{A}_j(y)).$$

Thus $b(y)$ is a vector of $\mathbb{R}^m$ and $\Delta(y)$ is a symmetric $m \times m$ matrix

Theorem 1 *The function f_r is twice continuously differentiable on $\widehat{Y}$. Actually, for all $y \in \widehat{Y}$ we have:*
a) $\nabla f_r(y) = b - rb(y)$;
b) $\nabla^2 f_r(y) = r\Delta(y)$;
c) the matrix $\Delta(y)$ is definite positive.

Proof: a) Denote by $(e_1, e_2, \cdots, e_m)$ the canonical basis of $\mathbb{R}^m$. Let $i \in \{1, \cdots, m\}$ and $z_i \in \mathbb{R}$, $z_i \neq 0$. Then,

$$\begin{aligned}\frac{f_r(y + z_i e_i) - f_r(y)}{z_i} &= b_i - \frac{r}{z_i}[\ln\det(B(y + z_i e_i)) - \ln\det(B(y))],\\ &= b_i - \frac{r}{z_i}[\ln\det(L(y)[I + z_i\widehat{A}_i]L^t(y)) - \ln\det(B(y))],\\ &= b_i - \frac{r}{z_i}\ln\det(I + z_i\widehat{A}_i(y)),\\ &= b_i - \frac{r}{z_i}\ln[1 + z_i\mathrm{trace}(\widehat{A}_i(y)) + z_i\varepsilon(z_i)].\end{aligned}$$

where the function ε is such that $\varepsilon(z) \to 0$ when $z \to 0$. Pass to the limit when $z_i \to 0$.

b) In the same manner, given $i, j \in \{1, \cdots, m\}$, let us consider

$$\frac{b_i(y + z_j e_j) - b_i(y)}{z_j} = \frac{-1}{z_j}[\mathrm{trace}(A_i[B^{-1}(y + z_j e_j) - B^{-1}(y)])].$$

But,

$$\begin{aligned}B^{-1}(y + z_j e_j) - B^{-1}(y) &= [B(y) + z_j A_j]^{-1} - B^{-1}(y),\\ &= [B(y)(I + z_j B^{-1}(y)A_j)\,]^{-1} - B^{-1}(y),\\ &= [(I + z_j B^{-1}(y)A_j)^{-1} - I\,]B^{-1}(y).\end{aligned}$$

Neglecting the second order terms in z_j, we obtain

$$\frac{b_i(y + z_j e_j) - b_i(y)}{z_j} \sim \mathrm{trace}(A_i B^{-1}(y)A_j B^{-1}(y)).$$

Pass to the limit when $z_j \to 0$. On the other hand the equality

$$\mathrm{trace}(B^{-1}(y)A_i B^{-1}(y)A_j) = \mathrm{trace}(\widehat{A}_i(y)\widehat{A}_j(y))$$

is immediate.

c) Let $d \neq 0$. Next, let $M = \sum_{i=1}^m d_i\widehat{A}_i(y)$. Then (H1) implies $M \neq 0$. On the other hand,

$$\langle\nabla^2 f_r(y)d, d\rangle = r\mathrm{trace}(\sum_{i,j} d_i d_j \widehat{A}_i(y)\widehat{A}_j(y)) = r\mathrm{trace}(M^2) > 0,$$

from what we deduce that the matrix $\nabla^2 f_r(y)$ is positive definite. □

Since f_r is strictly convex, (D_r) has at most one optimal solution.

3.2 (D_r) has one unique optimal solution

Because the convex function f_r takes the value $+\infty$ on the boundary of its domain and is differentiable on the interior, it is lower semi-continuous. In order to prove that (D_r) has one optimal solution, it suffices to prove that the recession cone of f_r is reduced to the origin. Before that, we show the following result:

Proposition 1 *$d = 0$ whenever $b^t d \leq 0$ and $\sum_{i=1}^m d_i A_i \in K$.*

Proof: Assume that $d \neq 0$, $b^t d \leq 0$ and $C = \sum_{i=1}^m d_i A_i \in K$. Then (H1) implies $C \neq 0$. Let some $\widehat{X} \in \widehat{F} \subset \widehat{K}$, such $\widehat{X}$ exists in view of assumption (H2). Then,

$$0 < \langle C, \widehat{X} \rangle = \sum_{i=1}^{m} d_i \langle A_i, \widehat{X} \rangle = b^t d.$$

The proposition is proved. □

Theorem 2 *$d = 0$ if $[f_r]_\infty(d) \leq 0$.*

Proof: Fix some $y \in \widehat{Y}$, such y exists in view of assumption (H2). The recession function $[f_r]_\infty$ of f_r is defined as

$$(f_r)_\infty(d) = \lim_{t \to +\infty} [\, \xi(t) = \frac{f_r(y + td) - f_r(y)}{t} \,].$$

Let $B = B(y) = \sum_{i=1}^m y_i A_i - C$, B is a positive definite symmetric matrix, there exists a non singular lower triangular matrix L such that $B = LL^t$. Given d, set $H(d) = \sum_{i=1}^m d_i A_i$. Then, for any t such that the matrix $B + tH(d)$ is positive definite,

$$\begin{aligned} \xi(t) &= b^t d - rt^{-1}[\ln \det(B + tH(d)) - \ln \det(B)], \\ &= b^t d - rt^{-1}[\ln \det(I + tE(d)] \end{aligned}$$

where $E(d) = L^{-1} H(d) (L^{-1})^t$. We deduce that ,

$$\xi(t) = \begin{cases} b^t d - rt^{-1} \ln \det\,(I + tE(d)) & \text{if } I + tE(d) \in \widehat{K}, \\ +\infty & \text{otherwise.} \end{cases}$$

The condition $[f_r]_\infty(d) \leq 0$ is therefore equivalent to say that $H(d)$ is positive semi-definite (hence $E(d)$ is also positive definite) and

$$b^t d \leq r \lim_{t\to\infty} \frac{1}{t} \ln\det\left(I + tE(d)\right) = r \lim_{t\to\infty} \sum_{i=1}^{n} \frac{1}{t} \ln(1 + t\lambda_i(d)) = 0,$$

where by $\lambda_i(d)$ we denote the eigenvalues of $E(d)$. Pass to the limit and apply Proposition 1. □

We denote by $y(r)$ or y_r the unique optimal solution of (D_r).

3.3 When $r \to 0$.

Next, we turn our interest in the behavior of the optimal value $m(r)$ and the optimal solution $y(r)$ of (D_r) for $r \to 0$. For that, let us introduce the function $h : \mathbb{R}^m \times \mathbb{R} \to (-\infty, +\infty]$ defined by

$$h(y,t) = \begin{cases} b^t y - \ln\det[\sum_{i=1}^{m} y_i A_i - tC] & \text{if } \sum_{i=1}^{m} y_i A_i - tC \in \widehat{K}, \\ +\infty & \text{otherwise.} \end{cases}$$

It is easily shown that h is convex and lower semi-continuous. Next, consider the function $\phi : \mathbb{R}^m \times \mathbb{R} \times \mathbb{R} \to (-\infty, +\infty]$ defined by

$$\phi(y,t,r) = \begin{cases} rh(r^{-1}y, r^{-1}t) & \text{if } r > 0, \\ h_\infty(y,t) & \text{if } r = 0, \\ +\infty & \text{if } r < 0. \end{cases}$$

Then, ϕ is also lower semi-continuous and convex, see for instance Rockafellar [8]. Next, define $f : \mathbb{R}^m \times \mathbb{R} \to (-\infty, +\infty]$ by

$$f(y,r) = \phi(y,1,r)$$

f is also convex and lower semi-continuous. By construction,

$$f(y,r) = \begin{cases} f_r(y) & \text{if} \quad r > 0, \\ b^t y & \text{if} \quad r = 0,\ y \in Y, \\ +\infty & \text{otherwise.} \end{cases} \tag{2}$$

Define $m : \mathbb{R} \to (-\infty, +\infty]$ by $m(r) = \inf_y[f(y,r) : y \in \mathbb{R}^m]$. This function is convex. Furthermore $m(0) = m_d$ and $m(r)$ is the optimal value of (D_r) when $r > 0$. It is clear that for $r > 0$

$$m(r) = f_r(y(r)) = f(y(r), r)$$

and

$$0 = \nabla f_r(y(r)) = \nabla_y f(y(r), r) = b - rb(y_r).$$

Theorem 3 *The functions m and y are continuously differentiable on $(0, +\infty)$. We have, for all $r > 0$,*

$$r\Delta(y_r)y'(r) - b(y_r) = 0,$$

$$m'(r) = n + n\ln(r) - \ln(\det(B(y_r)).$$

Moreover,

$$m_d = m(0) \leq b^t y(r) \leq m_d + nr. \tag{3}$$

Proof: Let $\bar{r} > 0$, $\nabla_y f(y(\bar{r}), \bar{r}) = 0$ because $y(\bar{r})$ is an optimal solution of $(D_{\bar{r}})$. The function f is twice continuously differentiable on $\widehat{Y}\times]0, +\infty[$ and the matrix $\nabla^2_{yy} f(y(\bar{r}), \bar{r})$ is positive definite. Applying the implicit function theorem to the equation $0 = T(y, r) = \nabla_y f(y, r)$ at the point $(y(\bar{r}), \bar{r})$ we deduce that in a neighborhood of $\bar{r}$ the function y is continuously differentiable and

$$\nabla^2_{yy} f(y_r, r)y'(r) - b(y_r) = 0.$$

Since $m(r) = f(y(r), r)$ and y is continuously differentiable on $(0, \infty)$, m is also continuously differentiable and

$$\begin{aligned} m'(r) &= f'_y(y(r), r)y'(r) + f'_r(y(r), r) \\ &= n + n\ln(r) - \ln[\det(B(y_r)]. \end{aligned}$$

Next, because the function m is convex,

$$m(0) \geq m(r) + (0 - r)m'(r)$$

from what we obtain,

$$+\infty > m_d = m(0) \geq b^t y(r) - nr > -\infty.$$

On the other hand $y_r \in \widehat{Y} \subset Y$ and therefore $b^t y(r) \geq m_d$. □

Let us denote by S_d the set of optimal solutions of (D), we know that this set is closed convex bounded and not empty. The distance of a point y to the set S_D is defined as

$$d(y, S_d) = \inf_z [\, \|y - z\| \, : \, z \in S_d \,].$$

The following result concerns the behavior of y_r and $m(r)$ when $r \to 0$.

Theorem 4 *Assume that $r \to 0$, then $d(y_r, S_d) \to 0$ and $m(r) \to m_d$.*

Proof: Let us consider the multivalued map S defined on $\mathbb{R}$ by

$$S(r) = \{y \in Y \, : \, b^t y \leq m_d + nr\}.$$

Its graph is closed, $S(r) = \emptyset$ if $r < 0$. If $r > 0$, $\emptyset \neq S(0) = S_d \subset S(r)$, $y_r \in S(r)$ and $S(r)$ is a closed convex set. The recession cone of $S(r)$, $r > 0$, coincides with the recession cone of $S(0)$. Hence $S(r)$ is compact because $S(r)$ is so. We deduce that the multivalued map S is upper semi-continuous (USC) on $[0, +\infty)$ because compact-valued with a closed graph. Since $y_r \in S(r)$, then $d(y_r, S(0)) \to 0$ when $r \to 0$.

It remains to prove that $m(r) \to m_d = m(0)$ when $r \to 0$. Since the function m is convex, it is enough to prove that it is lower semi-continuous at 0. We proceed by contradiction, if not there exist $\lambda < m_d$ and a sequence $\{r_k\}$ of positive numbers converging to 0 such that $m(r_k) < \lambda$. Let $y_k = y(r_k)$. Then there exist $\bar{y} \in S_d$ and a sub-sequence $\{y_{k_l}\}$ converging to $\bar{y}$. Since the function f is lower semi-continuous on $\mathbb{R}^m \times \mathbb{R}$ and $f(\bar{y}, 0) = m_d > \lambda$, one has for l large enough

$$\lambda > m(r_{k_l}) = f(y_{k_l}, r_{k_l}) > \lambda$$

which is not possible. □

4 The Newton descent direction and the line-search

Due to the presence of the barrier function, the problem (D_r) can be considered as unconstrained. This problem will be solved via a classical descent algorithm. Because the function f_r takes the value ∞ on the boundary of Y, the iterates will stay in $\widehat{Y}$. Thus, the method that we propose is an interior point method.

Assume that our current iterate is $y \in \widehat{Y}$. For descent direction d at y, we take the solution of the linear system

$$[\nabla^2 f_r(y)]d = -\nabla f_r(y).$$

According to Theorem 1, the linear system is equivalent to the system

$$\Delta(y)d = b(y) - \frac{1}{r}b \tag{1}$$

with $B(y)$, $b(y)$ and $\Delta(y)$ defined as in subsection 3.1. The matrix $\Delta(y)$ being definite positive, the linear system (1) can be efficiently solved via a Cholesky decomposition. Of course, we assume $\nabla f_r(y) \neq 0$ (if not the optimum is reached). It follows that $d \neq 0$.

The next step in the algorithm consists in the choice of $\bar{t} > 0$ giving a significant decrease of the function f_r on the half line $y + td$, $t > 0$. Then, the next iterate will be taken equal to $y + \bar{t}d$. To do that, we consider the function

$$\begin{aligned} \theta(t) &= \frac{1}{r}[f_r(y + td) - f_r(y)], \quad y + td \in \widehat{Y}, \\ \theta(t) &= \frac{t}{r}b^t d - \ln\det(B(y + td)) + \ln\det(B(y)). \end{aligned}$$

Since $\nabla^2 f_r(y)d = -\nabla f_r(y)$ one has

$$d^t \nabla^2 f_r(y)d = -d^t \nabla f_r(y) = d^t b(y) - r d^t b.$$

In order to simplify the notation, y and d staying fixed in the following, we set

$$B = B(y) = \sum_{i=1}^{m} y_i A_i - C \quad \text{and} \quad H = \sum_{i=1}^{m} d_i A_i.$$

Since B is symmetric and positive definite, there exists a lower triangular matrix L such that $B = LL^t$. Next, we set

$$E = L^{-1}H[L^{-1}]^t.$$

Since $d \neq 0$, assumption (H1) implies $H \neq 0$ from what we have $E \neq 0$.

With this notation, for all $t > 0$ such that $I + tE$ is positive definite,

$$\theta(t) = t[\mathrm{trace}(E) - \mathrm{trace}(E^2)] - \ln\det(I + tE). \tag{2}$$

Denote by λ_i the eigenvalues of the symmetric matrix E, then

$$\theta(t) = \sum_{i=1}^{n}[t(\lambda_i - \lambda_i^2) - \ln(1 + t\lambda_i)], \quad t \in [0, \widehat{t}) \tag{3}$$

where

$$\widehat{t} = \sup[\, t \,:\, 1 + t\lambda_i > 0 \ \text{ for all } i\,] = \sup[\, t \,:\, y + td \in \widehat{Y}]. \tag{4}$$

Observe that $\widehat{t} = +\infty$ if E is positive semi-definite and $0 < \widehat{t} < +\infty$ if not. It is clear that θ is convex on $[0, \widehat{t}[$, $\theta(0) = 0$ and

$$0 < \sum \lambda_i^2 = \theta''(0) = -\theta'(0).$$

Also $\theta(t) \to +\infty$ when $t \to \widehat{t}$. It follows that there exists one unique t_{opt} such that $\theta'(t_{opt}) = 0$, θ reaches its minimum in this point.

Unfortunately, there is no explicit formula giving t_{opt} and solving the equation by iterative methods needs successive computations of the functions θ and θ'. These computations have a high numerical cost because the expression of θ in (2) contains a determinant not easily handled and (3) needs the knowledge of the eigenvalues of E, a difficult numerical problem. This leads to think of alternative approaches.

Once E is computed, it is easy to compute the two following quantities

$$\mathrm{trace}(E) = \sum_i e_{ii} = \sum_i \lambda_i \quad \text{and} \quad \mathrm{trace}(E^2) = \sum_{i,j} e_{ij}^2 = \sum_i \lambda_i^2.$$

In section 6, we take advantage of these data to propose lower bounds of $\widehat{t}$ and functions bounded from below by θ. Before, we look at some useful inequalities on a sample of numbers when the sum of the numbers and the sum of their squares are known.

5 Some useful inequalities

As usual in statistics, given a sample of n real numbers $x_1, x_2, \cdots, x_n$, we consider their arithmetic mean $\bar{x}$ and their standard deviation σ_x. These quantities are defined as follows:

$$\bar{x} = \frac{1}{n}\sum x_i \quad \text{and} \quad \sigma_x^2 = \frac{1}{n}\sum x_i^2 - \bar{x}^2 = \frac{1}{n}\sum (x_i - \bar{x})^2.$$

The following result is due to Wolkowicz-Styan [10], see also Crouzeix-Seeger [5] for additional results.

Proposition 2

$$\bar{x} - \sigma_x\sqrt{n-1} \leq \min_i x_i \leq \bar{x} - \frac{\sigma_x}{\sqrt{n-1}},$$

$$\bar{x} + \frac{\sigma_x}{\sqrt{n-1}} \leq \max_i x_i \leq \bar{x} + \sigma_x\sqrt{n-1}.$$

In the particular case where all x_i are positive, one deduces

$$n\ln(\bar{x} - \sigma_x\sqrt{n-1}) \leq \sum_{i=1}^{n} \ln(x_i) \leq n\ln(\bar{x} + \sigma_x\sqrt{n-1}),$$

where, by convention, $\ln(t) = -\infty$ if $t \leq 0$. The next result is still better.

Theorem 5 *Assume that $x_i > 0$ for $i = 1, 2, \cdots, n$. Then*

$$n\ln(\bar{x} - \sigma_x\sqrt{n-1}) \leq A \leq \sum_{i=1}^{n} \ln(x_i) \leq B \leq n\ln(\bar{x}),$$

with

$$A = (n-1)\ln(\bar{x} + \frac{\sigma_x}{\sqrt{n-1}}) + \ln(\bar{x} - \sigma_x\sqrt{n-1}),$$

and

$$B = \ln(\bar{x} + \sigma_x\sqrt{n-1}) + (n-1)\ln(\bar{x} - \frac{\sigma_x}{\sqrt{n-1}}).$$

Proof: If $\sigma_x = 0$, then $x_i = \bar{x}$ for all i and the inequalities hold. Assume $\sigma_x > 0$. Let us consider the two following problems where $\bar{x}$ and σ_x are fixed,

$$A = \inf_x [\sum_{i=1}^n \ln(x_i) \ : \ \sum_{i=1}^n (x_i - \bar{x}) = 0 \quad \text{and} \quad \sum_{i=1}^n (x_i - \bar{x})^2 = n\sigma_x^2],$$

$$B = \sup_x [\sum_{i=1}^n \ln(x_i) \ : \ \sum_{i=1}^n (x_i - \bar{x}) = 0 \quad \text{and} \quad \sum_{i=1}^n (x_i - \bar{x})^2 = n\sigma_x^2].$$

The second problem has always optimal solutions, the first problem has optimal solutions if $\bar{x} - \sigma_x\sqrt{n-1} > 0$ because Proposition 2 and in the other case $-\infty = A < \sum \ln(x_i)$.

Apply the first order necessary optimality condition: if x is optimal solution of one problem or the other one, there exist α and β such for all i

$$(x_i - \bar{x})^2 - \alpha(x_i - \bar{x}) + \beta = 0.$$

Thus each $(x_i - \bar{x})$ is a root of the equation

$$w^2 - \alpha w + \beta = 0.$$

Denote by a and b the two roots of this equation. The quantities $(x_i - \bar{x})$ divide into two parts, p equal to a, $n - p$ equal to b. From $\sigma_x \neq 0$ we deduce that $1 \leq p \leq n - 1$ and $a \neq b$. Hence,

$$0 = \sum_i (x_i - \bar{x}) = pa + (n - p)b,$$

and

$$n\sigma_x^2 = \sum_i (x_i - \bar{x})^2 = pa^2 + (n - p)b^2.$$

From what we deduce that either

$$a = \bar{x} + \sigma_x\sqrt{\frac{n-p}{p}} \quad \text{and} \quad b = \bar{x} - \sigma_x\sqrt{\frac{p}{n-p}}$$

or

$$a = \bar{x} - \sigma_x\sqrt{\frac{n-p}{p}} \quad \text{and} \quad b = \bar{x} + \sigma_x\sqrt{\frac{p}{n-p}}.$$

Denote by $h(p)$ and $k(p)$ the following quantities

$$h(p) = \frac{p}{n} \ln \left[\bar{x} + \sigma_x \sqrt{\frac{n-p}{p}} \right] + \frac{n-p}{n} \ln \left[\bar{x} - \sigma_x \sqrt{\frac{p}{n-p}} \right],$$

$$k(p) = \frac{p}{n} \ln \left[\bar{x} - \sigma_x \sqrt{\frac{n-p}{p}} \right] + \frac{n-p}{n} \ln \left[\bar{x} + \sigma_x \sqrt{\frac{p}{n-p}} \right].$$

Then,

$$\frac{A}{n} = \min_{p=1,\cdots,n-1} [\min[h(p), k(p)]] \quad \text{and} \quad \frac{B}{n} = \max_{p=1,\cdots,n-1} [\max[h(p), k(p)]].$$

But $h(p) = k(n-p)$ for any $p = 1, ..., n-1$ and therefore

$$\frac{A}{n} = \min_{p=1,\cdots,n-1} [h(p)] \quad \text{and} \quad \frac{B}{n} = \max_{p=1,\cdots,n-1} [h(p)]. \tag{5}$$

It is interesting to set $t(p) = \sqrt{\frac{p}{n-p}}$ and to consider the function

$$\gamma(t) = \frac{t^2}{t^2+1} \ln[\bar{x} + \sigma_x t^{-1}] + \frac{1}{t^2+1} \ln[\bar{x} - \sigma_x t].$$

Then,

$$\gamma'(t) = \frac{2t}{(1+t^2)^2} [\ln \frac{\bar{x} + \sigma_x t^{-1}}{\bar{x} - \sigma_x t}] - \frac{\sigma_x}{(1+t^2)} \left[\frac{1}{\bar{x} + \sigma_x t^{-1}} + \frac{1}{\bar{x} - \sigma_x t} \right].$$

The concavity of the function $t \to t^{-1}$ implies that

$$\ln(t+\delta) - \ln(t) < \frac{\delta}{2} [\frac{1}{t+\delta} + \frac{1}{t}] \qquad \forall\, t, \delta > 0,$$

from what we deduce that γ' is negative on $(0, \infty)$ and therefore γ is decreasing on this interval. It follows that

$$A = n\gamma(n-1) < B = n\gamma(\frac{1}{n-1}) < n \lim_{t \downarrow 0} \gamma(t) = n \ln(\bar{x}).$$

The remaining inequality is a straight consequence of the definition of A. □

6 Back to the step-size procedure

Let us go back to equations (3) and (4). We denote by $\bar{\lambda}$ and σ_λ the arithmetic mean and the standard deviation of the λ_i and by $\|\lambda\|$ the euclidean norm of

the vector λ. Then, $\|\lambda\|^2 = n(\bar{\lambda}^2 + \sigma_\lambda^2) = \theta''(0) = -\theta'(0)$ and

$$\theta(t) = nt\bar{\lambda} - t\|\lambda\|^2 - \sum_{i=1}^{n} \ln(1 + t\lambda_i).$$

Our problem consists to find some $\bar{t} \in (0, \widehat{t})$ giving a significant decrease of the convex function θ.

We have said that the most natural choice, $\bar{t} = t_{opt}$ where $\theta'(t_{opt}) = 0$, presents numerical complications. It can be thought of a line-search by a method of Armijo-Goldstein-Price type but this line-search needs also several computations of functions θ and θ'. Nevertheless, if we decide for such a line-search, it is convenient to know, for lack of the upper-bound $\widehat{t}$ of the domain of θ which is numerically difficult to obtain, a lower-bound of $\widehat{t}$. Such a bound is issued from Proposition 2

$$\widehat{t}_1 = \sup\,[\,t\,:\,1 + t\beta_1 > 0\,] \quad \text{with} \;\; \beta_1 = \bar{\lambda} - \sigma_\lambda\sqrt{n-1}.$$

Another bound $\widehat{t}_2$ is due to the fact that $|\lambda_i| \leq \|\lambda\|$ for all i

$$\widehat{t}_2 = \sup\,[\,t\,:\,1 + t\beta_2 > 0\,] \quad \text{with} \;\; \beta_2 = -\|\lambda\|.$$

Then, $0 < \widehat{t}_2 \leq \widehat{t}_1 \leq \widehat{t} \leq +\infty$. As already said, the inequality $\widehat{t} \geq \widehat{t}_1$ is a consequence of Proposition 2. To prove that $\widehat{t}_1 \geq \widehat{t}_2$ it is enough to prove that $\|\lambda\|^2 \geq \beta_1^2$. This inequality is equivalent to

$$0 \leq (n-1)\bar{\lambda}^2 + \sigma_\lambda^2 + 2\sigma_\lambda\bar{\lambda}\sqrt{n-1} = (\bar{\lambda}\sqrt{n-1} + \sigma_\lambda)^2.$$

Another strategy consists in minimizing an upper-approximation $\widetilde{\theta}$ of θ. To be efficient, this approximation must be simple and close enough to θ. Here we require

$$0 = \widetilde{\theta}(0), \quad \|\lambda\|^2 = \widetilde{\theta}''(0) = -\widetilde{\theta}'(0).$$

Theorem 5 provides such an approximation: set $x_i = 1 + t\lambda_i$, then $\bar{x} = 1 + t\bar{\lambda}$ and $\sigma_x = t\sigma_\lambda$. Next, define

$$\theta_0(t) = \gamma_0 t - (n-1)\ln(1 + \alpha_0 t) - \ln(1 + \beta_0 t),$$

with

$$\gamma_0 = n\bar{\lambda} - \|\lambda\|^2, \;\; \alpha_0 = \bar{\lambda} + \frac{\sigma_\lambda}{\sqrt{n-1}} \text{ and } \quad \beta_0 = \beta_1 = \bar{\lambda} - \sigma_\lambda\sqrt{n-1}.$$

It is clear that θ_0 is convex, its domain is $[0, \widehat{t}_0)$ with $\widehat{t}_0 = \widehat{t}_1$ and

$$\theta(t) \leq \theta_0(t) \;\; \forall\, t \geq 0, \;\; \theta_0(0) = 0 \quad \text{and} \;\; \theta_0''(0) = -\theta_0'(0) = \|\lambda\|^2.$$

One can also thought of simpler functions than θ_0 involving only one logarithm. We consider functions of the following type

$$\widetilde{\theta}(t) = \widetilde{\gamma}t - \widetilde{\delta}\ln(1 + \widetilde{\beta}t), \;\; t \in [0, \widetilde{t})$$

where in order to fulfill the requirements

$$\|\lambda\|^2 = \widetilde{\delta}\widetilde{\beta}^2 = \delta\widetilde{\beta} - \widetilde{\gamma}, \qquad \widetilde{t} = \sup\,[\, t \,:\, 1 + t\widetilde{\beta} > 0\,].$$

Such functions are convex.

Of course $\widetilde{t} \leq \widehat{t}$ is required. In line with the lower-bounds $\widehat{t}_1$ and $\widehat{t}_2$, we consider the two functions θ_1 and θ_2 corresponding to β_1 and β_2. In the following result, we compare θ_0, θ_1 and θ_2 . As in other parts of the paper $\ln(r) = -\infty$ if $r \leq 0$.

Proposition 3 *θ_i, $i = 0, 1, 2$, is strictly convex on $[0, \widehat{t}_i)$, $\theta_i(t) \to +\infty$ when $t \to \widehat{t}_i$. Furthermore, $\theta(t) \leq \theta_0(t) \leq \theta_1(t) \leq \theta_2(t) \leq +\infty$ for all $t > 0$.*

Proof: The first part is immediate. The inequality $\theta(t) \leq \theta_0(t)$ is a straight consequence of Theorem 5. Set $\nu(t) = \theta_1 - \theta_0$. Because $\beta_0 = \beta_1$ and $\alpha_0 \geq \beta_0$ one has for $t > 0$

$$\nu''(t) = \frac{\delta_1\beta_1^2 - \beta_0^2}{(1+\beta_0 t)^2} - \frac{(n-1)\alpha_0^2}{(1+\alpha_0 t)^2} = \frac{(n-1)\alpha_0^2}{(1+\beta_0 t)^2} - \frac{(n-1)\alpha_0^2}{(1+\alpha_0 t)^2} \geq 0.$$

Because $\nu(0) = \nu'(0) = 0$, one deduces that $\nu(t) \geq 0$ for $t > 0$.

Next, set $\mu(t) = \theta_2 - \theta_1$. Then, $\mu(0) = \mu'(0) = 0$ and

$$\mu''(t) = \|\lambda\|^2[\frac{1}{(1+\beta_2 t)^2} - \frac{1}{(1+\beta_1 t)^2}] \geq 0.$$

Here again $\mu(t) \geq 0$ for all $t > 0$. □

We deduce that the function θ_i reaches its minimum in one unique value $\bar{t}_i$ which is the root of the equation $\theta_i'(t) = 0$. For $i = 1, 2$ one has

$$\bar{t}_i = \frac{\delta_i}{\gamma_i} - \frac{1}{\beta_i} \quad \text{and} \quad \theta_i(\bar{t}_i) = \frac{\|\lambda\|^2}{\beta_i} + \frac{\|\lambda\|^2}{\beta_i^2} \ln(1 - \beta).$$

In particular,

$$\bar{t}_2 = \frac{1}{1 + \|\lambda\|} \quad \text{and} \quad \theta_2(\bar{t}_2) = -\|\lambda\| + \ln(1 + \|\lambda\|).$$

The solution of $\theta_0'(t) = 0$ leads to the equation $t^2 - 2bt + c = 0$ with

$$b = \frac{1}{2}\left(\frac{n}{\gamma_0} - \frac{1}{\alpha_0} - \frac{1}{\beta_0}\right) \quad \text{and} \quad c = -\frac{\|\lambda\|^2}{\alpha_0 \beta_0 \gamma_0},$$

whose the two roots are $t = b \pm \sqrt{b^2 - c}$. For $\bar{t}_0$ we take the root which belongs to the interval $(0, \widehat{t}_0)$ (there is only one).

Thus, the three values $\bar{t}_0$, $\bar{t}_1$ and $\bar{t}_2$ are explicitly computed. It is clear that

$$\theta(\bar{t}_2) \leq \theta_2(\bar{t}_2), \quad \theta(\bar{t}_1) \leq \theta_1(\bar{t}_1) \leq \theta_1(\bar{t}_2) \leq \theta_2(\bar{t}_2)$$

and

$$\theta(\bar{t}_0) \leq \theta_0(\bar{t}_0) \leq \theta_0(\bar{t}_1) \leq \theta_1(\bar{t}_1) \leq \theta_2(\bar{t}_2).$$

7 Description of the algorithm:

Initialization: One decides for a step-size strategy and we choose the parameters $\varepsilon > 0$, $r > 0$, $\rho > 0$, $\sigma \in (0, 1)$. We start with some $y \in \widehat{Y}$.

Main step :

(a) Compute $B = B(y)$ and L such that $LL^t = B$.

(b) Compute $g = b - rb(y)$ and $H = r\Delta(y)$.

(c) Solve the equation $Hd = -g$. Compute E, trace(E) and trace(E^2).

(d) Compute $\bar{\lambda}$ and $\bar{\sigma}_\lambda$.

(e) Obtain $\bar{t}$ using the step-size strategy. Take $\bar{y} = y + \bar{t}d$.

(f) If $|b^t y - b^t \bar{y}| > \rho nr$, do $y = \bar{y}$ and go to **(a)**.

(g) If $nr > \varepsilon$, do $y = \bar{y}$, $r = \sigma r$ and go to **(a)**.

(h) **Stop:** $\bar{y}$ is an approximate solution of the problem (D).

As said previously, the optimal solution of problem (D_r) is only one approximate solution of problem (D), more r is close to 0, more the approximation is good. Unfortunately, more r is close to 0, more (D_r) is badly conditioned. It is the reason why we use in the first iterations of the algorithm large r instead of dealing directly with a value of r such that $nr < \varepsilon$. The reason for the updating of r is the following: if $y(r)$ is the exact solution of problem (D_r), then $b^t y(r) \in [m_d, m_d + nr]$, it is wasting time to continue iterations on (D_r) when $|b^t y - b^t \bar{y}| \leq \rho nr$, with ρ near 1. For ρ one can consider for instance the values 0.5, 1, 2 , 3. For σ, we can take for instance the values 0.1, 0.25, 0.5. We describe now four different strategies for the step-size:

- **Strategy Ls**: A classical line-search of Armijo-Goldstein-Price type.
- **Strategy S**$_i$, $i = 0, 1, 2$: $\bar{t} = \bar{t}_i$ with $\bar{t}_i$ defined as in the last section.

8 Numerical experiments

The computations have been performed on a D 810 station with Delphi 5.

8.1 Example cube

$n = 2m$, C is the $n \times n$ identity matrix, $b = (2, ..., 2)^t \in \mathbb{R}^m$ and the entries of the $n \times n$ matrix A_k, $k = 1, \cdots m$, are given by:

$$A_k[i,j] = \begin{cases} 1 & \text{if } i = j = k & \text{or } i = j = k + m, \\ a^2 & \text{if } i = j = k + 1 & \text{or } i = j = k + m + 1, \\ -a & \text{if } i = k, j = k + 1 & \text{or } i = k + m, j = k + m + 1, \\ -a & \text{if } i = k + 1, j = k & \text{or } i = k + m + 1, j = k + m, \\ 0 & \text{otherwise.} \end{cases}$$

$a \in \mathbb{R}$ is given.

Test 1: $(m, n) = (50, 100)$ and $a = 0$. Then, it is known that the vector $y = (1, ..., 1)^t \in \mathbb{R}^m$ is the optimal solution and $y_0 = (1.5, ..., 1.5)^t \in \mathbb{R}^m$ is feasible. We take for parameters in the algorithm $\rho = 1$, $\sigma = 0.125$, $r_0 = 0.3$, $\varepsilon = 0.1$ and for initial point y_0 . The following array describes the results.

Strategy	Computational time	Number of iterations
S_0	34 s	3
S_1	34 s	4
S_2	11 mn	25
Ls	dvg	dvg

dvg means that the algorithm does not terminate within a finite time.

Test 2: In this test, the data are the same as in the first test, except $\rho = 2$ in place of $\rho = 1$.

Strategy	Computational time	Number of iterations
S_0	33 s	3
S_1	34 s	4
S_2	8 mn	18
Ls	dvg	dvg

The results of two tests show that the strategies S_2 and Ls do not compete with S_0 and S_1. In the next experiments, we continue only the tests with S_0 and S_1.

Test 3: Same data as in test 1 except $C = -2I$ in place of $C = I$. We start the algorithm with $y_0 = (0, ..., 0)^t$ and $\rho = 1$.

Strategy	S_0		S_1	
	a=2	a=5	a=2	a=5
computational time	165 s	80 s	180 s	120
iteration	10	5	12	13

Test 4: Same data as in test 3 except $\rho = 2$ in place of $\rho = 1$.

Strategy	S_0		S_1	
	a=2	a=5	a=2	a=5
computational time	130 s	50 s	135 s	56 s
iteration	7	3	10	5

References

[1] F. Alizadeh, Interior point methods in semi-definite programming with application to combinatorial optimization, SIAM Journal on Optimization 5, 1995, 13–55.

[2] F. Alizadeh, J.-P. Haberly, and M.-L. Overton, Primal-dual interior-point methods for semi-definite programming, convergence rates, stability and numerical results, SIAM Journal on Optimization 8, 1998, 746–768.

[3] D. Benterki, J.-P. Crouzeix, and B. Merikhi, A numerical implementation of an interior point method for semi-definite programming, Pesquisa Operacional 23–1, 2003, 49–59.

[4] J.F. Bonnans, J.-C. Gilbert, C. Lemaréchal, and C. Sagastizàbal, Numerical optimization, theoretical and practical aspects, Mathematics and Applications 27, Springer-Verlag, Berlin, 2003.

[5] J.-P. Crouzeix and A. Seeger, New bounds for the extreme values of a finite sample of real numbers, Journal of Mathematical Analysis and Applications 197, 1996, 411–426.

[6] M. Kojima, S. Shindoh, and S. Hara, Interior point methods for the monotone semi-definite linear complementarity problem in symmetric matrices, SIAM Journal on Optimization 7, 1997, 86–125.

[7] M. Overton and H. Wolkowicz, Semi-definite programming. Mathematical programming, Serie B, 77, 1997, 105–109.

[8] R.-T. Rockafellar, Convex analysis, Princeton University Press, New Jerzey, 1970.

[9] L. Vanderberghe and S. Boyd, Positive definite programming, SIAM Review 38, 1996, 49–95.

[10] H. Wolkowicz and G.-P.-H. Styan, Bounds for eigenvalues using traces, Linear Algebra and Application, 29, 1980, 471–506.

Extending Greedy Local Search Algorithm to Unsatisfiability and Unvalidity

Gilles Audemard*[1] and Laurent Simon[2]

[1] Université d'Artois, CRIL - CNRS, FRE2499,
Lens, F-62307
audemard@cril.univ-artois.fr,
[2] Université Paris-Sud, LRI - CNRS, UMR8623,
INRIA-Futurs, Orsay, F-91405
simon@lri.fr

Abstract. Despite tremendous progresses observed on complete search algorithms over the last years, local search algorithms for satisfiability testing are still the best methods for a large number of problems, including some structured problems. However, their intrinsic limit does not allow them to address UNSAT problems. Ten years ago, this question challenged the community without any answer: was it possible to use local search algorithm for UNSAT formulae? We present here an approach addressing this issue, that can beat the best resolution-based (non-DPLL based) complete methods. We define the landscape of the search by approximating the number of filtered clauses by resolution proof. Furthermore, we add high-level reasoning mechanism, based on Extended Resolution and Unit Propagation Look-Ahead to make this new and challenging approach possible. We extend here our approach to new and challenging questions : obtaining short proofs for UNSAT problems and build a real local-search algorithm for QBF.

1 Introduction

Over the last ten years, a lot of impressive progresses have been made in the practical solving of the satisfiability testing (SAT). All the methods that address this typical NP-Complete problem may be divided in two categories: Complete ones, usually based on the Davis, Logemann and Loveland procedure [1, 2], can prove that a formula is satisfiable or not. Incomplete solvers (also called *one-sided* solvers) only give an answer if the instance has a given property (generally if it is satisfiable). Most of them, which are based on local search, perform a stochastic walk over the search space [3, 4]. Those methods give very good results on certain classes of problems, like random 3SAT problems [5].

In 1997, [6] challenged the local search community with a quest for an efficient local search algorithm for unsatisfiable formulae. We propose here the first successful approach, called GUNSAT, that pushes the local search formalism into resolution proofs search space. Because state-of-the-art complete SAT solvers are terribly efficient in practice, we only call *efficient* a solver, based on resolution, that can defeat state-of-the-art resolution-based solvers. For doing this, we propose to make a greedy walk among

* supported by ANR "Planevo" project n°JC05_41940.

the resolution search space in which, at each step of the local search algorithm, we try to compute a "*better*" neighbour proof, *i.e.* a proof which differs from the previous ones by at most two clauses, one added by resolution, and one that may have been removed. To find such a neighbour, we approximate the number of filtered models by the proof. This is achieved by a score given to all pairs of literals depending on their frequencies in the formula. As we will show it in our experimental investigation, the use of higher reasoning mechanism, based on Extended Resolution [7] and Unit Propagation Look-Ahead [8] is a key to make this new and challenging approach possible. Because resolution-based methods seems to be an efficient way for solving quantified boolean formulae (QBF), we believe that our algorithm may be the first step on two other challenging problems: build a real local-search algorithm for QBF and obtain short proofs for UNSAT problems and QBF ones.

The paper is organised as follows. We start by introducing some definitions and notations, then, in section 3, we discuss previous works done on local search for unsat problems, and we present our new approach. Before conclude, we provide some experiments.

2 Preliminaries and definitions

Let $V = \{x_1, \ldots, x_n\}$ be a set of *boolean variables*, a *literal* l_i is a variable x_i or its *negation* $\neg x_i$. A *clause* is a disjunction of literals $c_i = l_1 \vee l_2 ... \vee l_{n_i}$. A *unit* clause (or *mono-literal*) is a clause restricted to a single literal. A formula Σ is in *conjunctive normal form* (CNF) if Σ is a conjunction of clauses $\Sigma = c_1 \wedge c_2 ... \wedge c_m$. The SAT decision problem is defined as follows: given a formula Σ in CNF, is there an assignment of the variables V Σ such that Σ is satisfied, *i.e.* all the clauses of Σ are satisfied?

2.1 Local Search for SAT problems

Local search algorithms for SAT problems use a stochastic walk over total interpretations of Σ (*i.e.* all variables are assigned). At each *step* (or *flip*), they try to reduce the number of unsatisfiable clauses (under certain conditions). The next total interpretation is chosen among the neighbours of the current one (they differ only on one literal value). After a given number of steps, a restart is done to jump into an other part of the landscape (*escape phase*). We recall, algorithm 1, the general scheme of local search algorithms for SAT.

A lot of improvements have been proposed in the literature, like *tabu search* or the use of different heuristics to find the next interpretations. The interested reader may refer to [4] for a detailed coverage of all methods.

2.2 Resolution Proofs

The resolution rule allows to deduce a new clause (called resolvent) with the help of two clauses. It plays a central role in many works based on clausal representations of boolean formulae, like preprocessing steps [9].

Algorithm 1: General scheme of a local search SAT solver

```
Data   : Σ a CNF formula
Result : SAT if a model is found, UNKNOWN otherwise
begin
    for i=1 to MaxTries do
        Choose a random interpretation I;
        for j=1 to MaxFlips do
            if I is a model of Σ then return SAT ;
            I = neighbour(I);
        end
    end
    return UNKNOWN ;
end
```

Definition 1 ([7]). *Let $c_1 = (x \vee a_1 \vee a_2 \vee \ldots a_n)$ and $c_2 = (\neg x \vee b_1 \vee b_2 \vee \ldots b_m)$ be two clauses. The clause $c = (a_1 \vee a_2 \vee \ldots a_n \vee b_1 \vee \ldots b_m)$ is called the* resolvent *of the clauses c_1 and c_2 by performing a* resolution *on the variable x. We note $c = c_1 \otimes c_2$ this rule.*

One may apply the resolution rule many times including on resolvent clauses to derive new clause. The obtained clause is *proved* by resolution:

Definition 2 (Resolution proof [7]). *Let Σ be a CNF formula. A resolution proof of a clause c is a succession of clause $P = c_1, c_2, \ldots, c_k$ such that $c = c_k$ and for all $i \leq k$ one of the following conditions holds :*

- $c_i \in \Sigma$
- $\exists c_m, c_n \in P$ *($m < i$ and $n < i$) such that* $c_i = c_m \otimes c_n$

If the empty clause ($\perp$) may be proved by resolution then the formula Σ is unsatisfiable: The resolution proof system is complete for refutation. Restrictions on general resolution exist and may still answer both SAT and UNSAT questions (like *Ordered Resolution*). However, in practice, resolution-based solvers are more suitable to knowledge compilation problematics (preprocessing, prime implicates computation, ...) than to SAT checking. Directional Resolution [10] is one of the most famous restriction, based on the well known work of [11]. [12] has shown how a simple generalisation of the resolution rule, called multi-resolution, allowed to scale-up most of the resolution based solvers, essentially for compilation purpose. This two resolution-based solvers will help us measure the performances of GUNSAT. We call *efficient* a solver, based on the resolution rule, that can beat both of them.

Extended Resolution It is well known that the number of necessary clauses could be exponential to prove refutation by general resolution proof (even if no restrictions are imposed in the order of resolutions steps). This is the case for the well known pigeon hole problem [13], the Urquhart problems [14] and even the random problems [15]. We have here to notice that all those problems need *large* clauses (*i.e.* unbounded size) in

the resolution proof before producing $\bot$. On a lot of instances, allowing large clauses to be generated is essential.

However, it is striking to notice that as soon as one allows the introduction of new variables during the proof (like lemmas in theorem proving), then short proofs for the above problems exist. [16] introduced this principle as the *extended resolution* rule. If Σ is a formula, one application of this rule allows us to consider $\Sigma \wedge (e \Leftrightarrow l_1 \vee l_2)$ instead, where e is a fresh variable (not appearing in Σ) and l_1 and l_2 are literals from Σ. Even if this rule looks very simple, there is no proof system known to be stronger than extended resolution. Despite its simplicity and its theoretical interests, no practical applications of extended resolution are used in modern SAT solvers. Some works implicitly use a restriction of it, like symmetry breaking algorithm based on the introduction of new variables. The work of [12] can also be viewed as a restriction of this rule, where nodes of the graph representation of the formula may be viewed as new propositional variables. However, at the end, the whole problem remains: even if this rule looks simple, how to choose the pair of literals to extend?

3 Local search for UNSAT problems

After a discussion of previous works, we present GUNSAT, our new local search algorithm for unsat checking.

3.1 Previous Work

Among the ten challenges proposed ten years ago [6], not so much work has been done to address the fifth challenge: *"design a practical stochastic local search procedure for proving unsatisfiability"*.

Hybrid solvers Almost all previous works on incomplete algorithms for unsatisfiability has been done on hybrid solvers only. Some methods use a classical local search solver but record *nogoods*, based on resolvent. However, in the general case, those methods improve performance on satisfiable problems only [17, 18].

With the new interest in QBF solving and the relative failure of direct DPLL extensions on them, some effort have been done on finding new methods, based on resolution or local search. The solver WalkQSAT [19] has two distinct phases: the first one is a complete algorithm based on DPLL for QBF; the second one use local search to quickly find models. This incomplete solver can prove either the validity and the invalidity of a QBF formula, but may never prove it.

RANGER Ranger [20] is also a local search algorithm for UNSAT problems that may looks very similar to GUNSAT. In this recent work, arbitrary resolution is performed in order to generate a large number of the shortest possible clauses, as fast as possible. To avoid combinatorial explosion, they may remove clauses. As we will see after, GUNSAT has a more powerful reasoning mechanism and a finest heuristic to guide moves whereas Ranger is simpler but performs many more moves per second.

Algorithm 2: GUNSAT

```
Data   : Σ a CNF formula
Result : UNSAT if a derivation of ⊥ is found, UNKNOWN otherwise
begin
    for i=1 to MaxTries do
        for j=1 to MaxFlips do
            if 2-Saturation(Σ) returns UNSAT then
                return UNSAT ;
            if |Σ| > MaxSize then
                Remove-One-Clause(Σ)
            Add-One-Clause(Σ);
            Add-Extended-Variables(Σ);
            Simplify-Look-Ahead(Σ);
        end
        Replace Σ by all its vital clauses;
    end
    return UNKNOWN;
end
```

3.2 Skeleton of our local search algorithm

The solver GUNSAT has, for a large part, the same skeleton than classical local search algorithms. Intuitively, our algorithm will remove and add new clauses to the set of clauses Σ, trying to derive $\perp$ by the resolution rule. The whole problem is to ensure that our algorithm may be able to perform *greedy moves* in the search space. It is also fundamental to add a higher level reasoning mechanism to ensure good practical results. The GUNSAT algorithm, detailed in the later, is given in algorithm 2.

3.3 Inside GUNSAT

Estimation of the number of filtered clauses Given a set of clauses Σ built on n variables, we want to approximate the total number of its potential models that are directly filtered out (over the 2^n potential ones). The measure must be based on explicit information only, and the computational effort to obtain it must be polynomially bounded. Otherwise, one may need to exactly count the number of its models, which is clearly harder than the coNP-hard problem we are initially facing.

Let $c_i = l_1 \vee l_2 ... \vee l_{n_i}$ be a clause in the formula Σ. For any c_i s.t. $n_i \geq 1$, it is clear that the clause filter out 2^{n-n_i} of potential models.

Our first measure (*depth 0* of the approximation) estimates the number of filtered models considering clauses independently. Each clause c_i of length n_i weights $w_0(n_i) = 2^{n-n_i}$. The whole weight of the proof may then be measured as $\Sigma_{i=1}^{n} w_0(n_i)$. It is however clear that filtered models are not independent and this measure gives a very inaccurate indication of the *quality* of the current proof. It remains at trying to produce as many short clauses as possible.

At depth 1 of the approximation, we fix the granularity to literals, by maintaining the estimation of the number of filtered models for each literal. For each literal l, we may

sum its weight in all clauses where it occurs by considering that a clause c_i of length $n_i \geq 1$ filter out 2^{n-n_i} of the models that contain $\neg l$ (over its 2^{n-1} potential models). If we suppose that all these filtered models are separately and equally distributed over all literals in the clause, which is a strong assumption on which our estimation is based, then the clause c_i allows literal l to filter out $w_1(n_i) = \frac{2^{n-n_i}}{n_i}$ of the models containing $\neg l$. Even through this is a refinement of depth 0, it is clear that if l occurs in the two clauses $l \wedge q$ and $l \wedge \neg q$ then this scoring scheme is particularly inappropriate. In some way, it is important to take the locality of common variables into account.

This is what approximation of depth 2 does. The granularity is here pairs of literals. For a clause c_i of length n_i, the 2^{n-n_i} filtered models are supposed as equally distributed over the $n_i.(n_i-1)/2$ pairs of literals occurring in c_i. Each pair (l_1, l_2) appearing in c_i is credited a weight of $w_2(n_i) = \frac{2^{n-1-n_i}}{n_i.(n_i-1)}$. The score of a pair of literal (l_1, l_2) is defined as the sum of its weights in all clauses and noted $S(l_1, l_2)$. The score $S(c)$ of a clause c is the sum of the scores of all the pairs of literals it contains. In the remaining sections, w and S may be used without indices for depth 2.

The deeper the approximation is, higher the cost to maintain it. We may refine our measurement by considering triplet of literals, but any explicit representation of all possible triplets of literals may be quickly unpracticable on realistic UNSAT benchmarks.

Remark 1 (Clauses scoring). If a given clause c_i has a score $S(c_i) \simeq n_i.(n_i-1).w_2(n_i)/2$ then c_i is nearly the only one that filter the models composed by the negation of its literals. Even if such a clause is long, it should be kept in the proof. At the opposite, if $S(c_i) \gg n_i.(n_i-1).w_2(n_i)/2$ then there is a little hope that this clause is from great importance in the current proof.

What are Greedy Moves? We now have a scoring scheme for pairs and clauses. If we try to improve the total sum of clauses scores over Σ, then our algorithm will look for large clauses in which a lot of frequent pairs occur. This is the completely opposite goal.

Instead of trying to improve a measurement over the whole proof, we'll focus on *quadruplets* of pairs only. In order to derive $\perp$, we need a step in our proof where l and $\neg l$ are in Σ. Because we perform 2-Saturation with Unit-Clause propagation at each step of our moves (see in the later), such a case never occurs. So, we have to find two literals l_1 and l_2 such that clauses $l_1 \vee l_2$, $\neg l_1 \vee l_2$, $l_1 \vee \neg l_2$ and $\neg l_1 \vee \neg l_2$ can be derived from Σ. In other words, we'll try to improve the scores of the four pairs built on the same variables, called *quadruplets* and noted $[x_1, x_2]$. Because we want to localise quadruplets where all pairs have a high score, we define the score $S_q([x_1, x_2])$ over a quadruplet as the sum of the squares of the scores of its pairs $S_q([x_1, x_2]) = S(l_1, l_2)^2 + S(\neg l_1, l_2)^2 + S(l_1, \neg l_2)^2 + S(\neg l_1, \neg l_2)^2$. Any move that enhance the score of one of the best scored quadruplets is thus a greedy move.

Neighbourhood Adding any clause $c' \notin \Sigma$ to Σ such that $\Sigma \vdash c'$ may define the neighbourhood. However, the deduction mechanism must be restricted to a correct but polynomially bounded and uncomplete method. Otherwise, one may reduce GUNSAT to a single call to any complete SAT solver to check if $\Sigma \vdash \perp$ and then add it to Σ.

We chose to use the 1-Resolution mechanism, noted $\vdash^1_{\text{Res}}$, which is defined as $\Sigma \vdash^1_{\text{Res}} c$ iff $c = c_1 \otimes c_2$ where c_1 and c_2 occur in Σ. Even if this rule is very simple, we are already facing a huge neighbourhood. To illustrate this, let's take a random k-SAT formula at a given ratio r, with n variables and $r \times n$ clauses. Before the first move, each literal occurs on average $k.r.n/2$ times, and $k^2.r^2.n^2/4$ clauses may be derived from 1-Resolution on it, which give for the whole formula $n^3.k^2.r^2/4$ potential clauses. For a 700 variables 3SAT formula at threshold ($r = 4.25$), we already have over 10^{10} potential moves to rank. Random moves in this huge neighbourhood may only lead to blind search for unsatisfiability, without any hope to beat ordered resolution-based algorithms like [10, 12].

Finding the bests Greedy Moves Any explicit exploration of the neighbourhood is impossible in practice. However, if we know in advance that we have to increase the score of one of the bests quadruplets, we may try to increase the score of any of one of its pairs.

Let $[x_1, x_2]$ be the best scored quadruplet in Σ. Increasing the score of any of its pairs (l_1, l_2) amounts to adding any new clause containing both l_1 and l_2. Short clauses will grant (l_1, l_2) a higher score, and should be preferred. However, the score of the new clause itself has to be taken into account. If we add a short clause with a high score, then this clause will probably be considered as useless in the next iteration of the algorithm. Thus, we are looking for a short clause with the lowest possible score to filter out new potential models.

When looking for this new clause with l_1 and l_2, we first try to localise the pivot variable on which we'll perform the resolution rule. If such a pivot p exists, then $S(l1, p) > 0$ and $S(\neg p, l_2) > 0$. We thus have to choose a clause c_p containing l_1 and p, and a clause $c_{\neg p}$ containing $\neg p$ and l_2. To prevent the new clause from having a high score, we only try to generate the new clause c from the two clauses having the lowest scores. Because of this restrictions, it is not always possible to produce a new clause c containing l_1 and l_2 (for instance, c may be subsumed by some clause of Σ or may simply be a tautology). If one pair score cannot be improved, then the other pairs of the same quadruplet are iteratively tried. If GUNSAT fails on all pairs of the best quadruplet, then the second quadruplet is tried and so on. This is what the call to `Add-One-Clause` does.

Removing *useless* clauses As we have seen, clause scoring will be preferred to a simple measurement of clause length to localise useless ones. This mechanism allows us to keep large clauses in Σ, especially if it is the only one to filter out a large number of pairs. However, it is essential to keep *vital* clauses in Σ. They ensure that we are preserving its unsatisfiability. Vital clauses are initial clauses, or any clause that previously subsumed another vital clauses. We also forced the algorithm to keep binary clauses. This step is performed by the call to `Remove-One-Clause`.

Finding a random start As described in GUNSAT skeleton, random start and general iteration of the algorithm only differ in the removing or not of clauses at each step.

During the random start initialisation, no clauses are removed (except any subsumed clause) in order to fill the proof up to `MaxSize` clauses.

3.4 Other refinements

It has been essential to add three powerful mechanisms to GUNSAT in order to make it competitive to other resolution-based reasoning systems. The first but essential refinement concerns subsumptions. Before adding a new clause, we perform forward/backward subsumption detection [21].

Binary clause saturation The power of binary clause saturation has been exploited with success in [22] for preprocessing purposes. At each step of the walk, binary clauses, related to our pairs of literals, have a special treatment in the call to `2-Saturation`. Each time a new binary clause is found in Σ, all resolutions between the set of binary clauses are performed to saturate Σ with it. In order to exploit their full power, an equivalency literal search is performed: if clauses $l_1 \vee \neg l_2$ and $\neg l_1 \vee l_2$ are found in Σ, then all occurrences of l_1 are replaced by l_2 and l_1 is tagged as a new potential extended variable to be used in the latter search.

While performing the binary clause saturation, the algorithm may find new unary clauses l_1. The literal l_1 is then propagated in the whole formula by unit propagation (all clauses containing l_1 are deleted, all clauses containing $\neg l_1$ are shortened). An inconsistency may be found at this step and then returned to the main algorithm.

Extended Resolution When the algorithm has tried to increase the score of a given pair of literals too many-times without any success, we chose to adopt a tricky solution that really pays: use extended resolution (ER) to artificially increase the score of this pair of literals.

The extended rule $e \Leftrightarrow l_1 \vee l_2$ is encoded by the three clauses $(\neg e \vee l_1 \vee l_2)$, $(e \vee \neg l_1)$ and $(e \vee \neg l_2)$. If necessary, the `Add-Extended-Variable` step may add a fresh variable by adding all those three clauses to Σ. Of course, we ensured that a pair of literals can only appear in at most one extended rule. As we'll see section 4, this very simple rule gives very good results in practice.

Look Ahead techniques The patented Stålmarck method [23], which gives good results in practice on some structured benchmarks, uses a powerful Look Ahead technique to detect equivalencies between literals until an inconsistency is found. It uses a reformulation of the initial formula, based on a set of *triplets* ($p \Leftrightarrow q \Leftrightarrow r$ and $p \Leftrightarrow q \Rightarrow r$ only). When formulae are initially written in CNF, the power of this method may be partially captured by unit-propogation lookahead (LH) [8].

To enhance the power of GUNSAT, we added LH techniques on pairs of literals. The four possible values of pairs are iteratively propagated in Σ, searching for more, implied, unit propagations. If any literal l of Σ is set to $\bot$ in all the four tries, then LH proved that $\Sigma \vdash \neg l$, and the unary clause $\neg l$ is added to Σ.

Restarting When GUNSAT fails to derive $\perp$ from Σ after `MaxFlips` steps, a restart is performed. All clauses, except binary ones and the set of *vital* clauses, are removed. To ensure that GUNSAT will explore another part of the search space, we added random number to cut ties of quadruplets having the same scores up to a given ϵ. Each restart is followed by a new random generation of quadruplets random numbers. All clauses containing at least one extended variables are deleted after each restart, including binary ones.

A particularity of our local search algorithm may be emphasised here: because of unit propagation, binary clause saturation and subsumption deletion, Σ may evolve from starts to restarts. Hopefully, Σ will only evolve to a simpler and simpler formula.

4 Experimental evaluation

Our results were obtained with a Xeon 3.4Ghz with 2GB memory. GUNSAT is a prototype written in java and a lot of optimisations are still planned (Java API `LinkedList` data structure are for instance used and may be greatly improved). In addition, our current implementation use an explicit representation of all pairs, which is a very costly operation. We are planning to manipulate only the Top-M best pairs of literals in our next versions.

Because of these two weak points, we selected instances with only a very restricted number of clauses and variables (*aim*, *xor* and *jnh*). Furthermore, like other local search algorithms, GUNSAT has a lot of parameters to tune and some of them may be improved in the near future. We chose to perform 50 restarts and to fix the number of flips to 6 times the initial number of clauses.

As we have previously said, our purpose is not here to compare GUNSAT to state-of-the-art SAT solvers. It is clear that conflict learning solvers like ZCHAFF [2] for structured instances and solvers like KCNFS [24] for random instances outperform GUNSAT. We rather want to compare it with resolution-based solvers.

We compared 4 versions of our solver: the basic one, the one with lookahead (LH) only, with extended resolution only (ER) and the one with both of them (LH+ER).

4.1 Structured instances

Table 1 reports the results obtained by GUNSAT on structured problems. We do not report the original DR performances [10]: over the 187 runs, only 47 finished without a memory allocation error and only 9 instances were solved (all `aim-50` and 1 `xor`). ZRes [12] was able to solve all `aim-50`, half `aim-100`, but failed to solve `aim-200` and `jnh`. It was able to solve all `xor` instances, because of its ability to handle large set of highly structured clauses.

It is clear that proposed refinements (LookAhead and Extended Resolution) and especially their combination provide a realistic local search solver for UNSAT problems.

4.2 Random instances

Table 2 reports results on random instances with different number of variables (from 50 to 70) and different ratio (4.25, 5 and 6). We only report results for ER+LH because

	basic		LH		ER		LH + ER	
	% S	T (F)	% S	T (F)	% S	T (F)	% S	T (F)
aim-50 (8)	12	2.14 (26)	100	1.41 (146)	60	15.14 (1749)	100	1.58 (142)
aim-100 (8)	10	36.83 (3954)	55	11.93 (923)	27	139.74 (4998)	97	49.37 (1726)
aim-200 (8)	0	-	60	63.62 (1098)	5	739.00 (9099)	85	201.29 (2009)
jnh (33)	6	0.95 (0)	57	8.48 (276)	18	68.15 (986)	62	4.21 (687)
xor (39)	0	-		-	1.5	308.83 (6932)	11	31.91 (5197)

Table 1. Results on structured instances. %S gives the percentage of solved instances (5 launches per instance). T means Time (in seconds, averaged of all successful runs), and F gives the averaged total number of flips.

		LH + ER	
V	R	% S	T (F)
50	4.25	58	60 (3880)
50	5.0	86	18 (1520)
50	6.0	97	5 (545)
60	4.25	35	126 (5785)
60	5.0	68	67 (3346)
60	6.0	92	16 (1094)
70	4.25	23	189 (6626)
70	5.0	51	187 (6193)
70	6.0	87	59 (2389)

Table 2. 3-SAT Random instances - Each category contains 100 instances, each instance is solved 50 times

all other versions of GUNSAT were only able to solve at most 1% of the instances. The solver DR cannot solve either these instances. ZRes solves only instances with 50 variables at the threshold in 250 seconds on average. [20] also reports bad results on random instances with RANGER.

Even on hard examples for resolution based algorithms, GUNSAT LH+ER showed promising results, in comparison with resolution based solvers and RANGER, especially for large clause/variable ratios.

One may also notice the very low number of flips in all the cases, which mean that the obtained proof length is short, in comparison with ZRes experimentations on random instances.

5 Conclusion and future work

We report the first good results on local search for unsatisfiability since [6] challenged the community. GUNSAT differs from classical local search algorithm in its high-cost

flip mechanism. Finding and adding good clauses, while maintaining the score of all pairs, is a costly – but necessary – task to prevent a blind local search. GUNSAT learns from start to restarts and may be able to prove unsatisfiability start after restarts, by suddenly finding a derived unit clause before starting again with this new high-valuable clause. We showed that LH techniques with ER are two key points to allow local search on resolution proofs. With both mechanisms, made possible by our scoring based on pairs of literals, GUNSAT was able to defeat state-of-the-art resolution based solvers.

Those good results should be relativized by the fact that GUNSAT cannot yet challenge state of the art DPLL solvers. However, those solvers have two main drawbacks: they hardly exhibit a proof of unsatisfiability of the initial formula that can be checked by a third-party trusted algorithm and they seem to reach their limit when extended to QBF solving. Resolution-based algorithms may be the next generation of efficient QBF solving, with a short proof checking made possible. In this new and still promising area, we believe that GUNSAT may take a good place.

References

1. Davis, M., Logemann, G., Loveland, D.: A machine program for theorem-proving. Communications of the ACM **5**(7) (July 1962) 394–397
2. Moskewicz, M., Madigan, C., Zhao, Y., Zhang, L., Malik, S.: Chaff : Engineering an efficient sat solver. In: Proceedings of DAC. (2001) 530–535
3. Selman, B., Levesque, H., Mitchell, D.: A New Method for Solving Hard Satisfiability Problems. In: Proceedings of AAAI. (1994) 440–446
4. Hoos, H., Stutzle, T.: Stochastic Local Search, Foundations and Applications. Morgan Kaufmann / Elsevier (2004)
5. LeBerre, D., Simon, L., eds.: Special Volume on the SAT 2005 competitions and evaluations. Volume 2. Journal on Satisfiability, Boolean Modeling and Computation (2006)
6. Selman, B., Kautz, H., McAllester, D.: Ten challenges in propositional reasoning and search. In: Proc. of IJCAI. (1997) 50–54
7. Robinson, J.: A machine-oriented logic based on the resolution principle. JACM **12** (1965) 23–41
8. LeBerre, D.: Exploiting the real power of unit propagation lookahead. In: Proceedings of SAT. (2001)
9. Eén, N., Biere, A.: Effective preprocessing in sat through variable and clause elimination. In: in proceedings of SAT. (2005) 61–75
10. Dechter, R., Rish, I.: Directional resolution: the davis-putnam procedure, revisited. In: proceedings of KR. (1994) 134–145
11. Davis, M., Putnam, H.: A computing procedure for quantification theory. Journal of the ACM (1960) 201–215
12. Simon, L., del Val, A.: Efficient consequence finding. In: proceedings of IJCAI. (2001) 359–365
13. Haken, A.: The intractability of resolution. Theorical Computer Science **39** (1985) 297–308
14. Urquhart, A.: Hard examples for resolution. JACM **34**(1) (1987) 209–219
15. Chvátal, V., Szemerédi, E.: Many hard examples for resolution. Journal of the ACM **35** (1988) 759–768
16. Tseitin, G.: On the complexity of derivation in propositional calculus. Studies in Constructive Mathematics and Mathematical Logic **2** (1970) 466–483

17. Fang, H., Ruml, W.: Complete local search for propositional satisfiability. In: proceedings of AAAI. (2004) 161–166
18. Shen, H., Zhang, H.: Another complete local search method for sat. In: in proceedings of LPAR. (2005) 595–605
19. Gent, I., Hoos, H., Rowley, A., Smyth, K.: Using stochastic local search to solve quantified boolean formulae. In: Proc.of CP. (2003) 348–362
20. Prestwich, S., Lynce, I.: Local search for unsatisfiability. In: Proceedings of SAT. (2006) 283–296
21. Zhang, L.: On subsumption removal and on the fly cnf simplification. In: proceedings of SAT. Volume 3569 of LNCS. (2005) 482–489
22. Bacchus, F.: Enhancing davis putnam with extended binary clause reasoning. In: proceedings of AAAI. (2002) 613–619
23. Stålmarck, G.: A system for determining propositional logic theorem by applying values and rules to triplets that are generated from a formula. US Patent 5,276,897; Canadian Patent 2,018,828; European Patent 0 403 545; Swedish Patent 467 076 (1994)
24. Dubois, O., Dequen, G.: A backbone-search heuristic for efficient solving of hard 3–sat formulae. In: Proceedings of IJCAI. (2001)