RESEARCH IN COMPUTING SCIENCE

ISSN: 1870-4069

Advances in Computer Science and Engineering

Sulema Torres
Itzamá López
Hiram Calvo
(Eds.)

Vol. 27

RCS
Research in Computing Science

Advances in Computer Science and Engineering

ISSN: 1870-4069

Instituto Politécnico Nacional (IPN)
Centro de Investigación en Computación (CIC)
Av. Juan de Dios Bátiz s/n esq. M. Othón de Mendizábal
Unidad Profesional "Adolfo López Mateos", Zacatenco
07738, México D.F., México

http://www.ipn.mx
http://www.cic.ipn.mx

Indexed in LATINDEX
Indexada en LATINDEX

Printing: 500
Tiraje: 500

Printed in México
Impreso en México

Preface

CORE 2007 was the 8th annual conference on Computing. CORE means nucleus, so the aim of this conference is to gather, in a single event, information of the state of the art in Computer Science and Engineering.

The CORE conference is distinguished because it is organized mainly by students, thus, giving them the opportunity to broaden their academical knowledge by means of their interaction with other institutions and researchers.

Total of 73 papers by 212 authors from 17 different countries where submitted for evaluation; see Tables 1 and 2. Each submission was reviewed by three independent members of the Editorial Board of the volume.

This volume contains revised versions of 26 papers, by 81 authors, selected for publication after thorough evaluation. The acceptance rate was 35.6%. In Table 1, the number of papers by country was calculated by the shares of all authors of the paper: e.g., if a paper has three authors: two from Mexico and one from USA, then we incremented the counter for Mexico by 0.66 (two authors of three) and the counter for USA by 0.33. Table 2 presents the statistics of papers by topics according to the topics indicated by the authors; note that a paper can be assigned more than one topic.

Table 1. Statistics of submissions and accepted papers by country / region

	Authors		Papers[1]	
Country/Region	Subm	Accp	Subm	Accp
Argentina	33	10	7	2
Brazil	4	3	1.15	0.9
Chile	3	3	1.25	1.25
China	8	-	4	-
Colombia	1	-	0.33	-
Cuba	12	3	3.99	0.99
France	2	2	0.53	0.53
Germany	1	-	0.25	-
India	1	-	1	-
Ireland	3	3	0.99	0.99
Italy	1	-	1	-
Japan	1	1	0.5	0.5
Korea, South	1	-	1	-
Mexico	134	52	47.65	17.32
Spain	6	4	2	1.5
United Kingdom	1	-	0.33	-
Total:	*212*	*81*	*73*	*26*

[1] Counted by authors: e.g., for a paper by 3 authors: 2 from Mexico and 1 from USA, we added ⅔ to Mexico and ⅓ to USA.

Table 2 Statistics of submissions and accepted papers by topic[2]

Topic	Submitted	Accepted
Algorithm Theory	5	3
Artificial Intelligence	32	10
Bioinformatics	1	-
Cellular Automata	1	1
Computer Architecture	1	-
Computer Vision	3	2
Control Systems	4	1
Data Mining	10	4
Database Systems	1	1
Digital Signal Processing	5	3
Distributed Systems	7	4
Evolutionary Algorithms	5	2
Formal Languages	2	2
Fuzzy Logic	4	2
Geoprocessing	1	-
High-Performance Computing	3	1
Information Security	1	-
Knowledge Representation	7	2
Multi-agent Systems	11	3
Natural Language Processing	7	4
Networks and Connectivity	5	2
Neural Networks	4	2
Operating Systems	1	-
Parallelism	1	-
Real Time Systems	2	1
Robotics	6	2
Scientific Computing	4	2
Semantic Web	4	1
Software Engineering	7	2
Web Design	6	1
Other	18	7

[2] According to the topics indicated by the authors. A paper may be assigned to more than one topic.

The papers are structured into the following nine sections:

- Knowledge Acquisition and Neural Networks
- Fuzzy Logic
- Optimization
- Image Processing
- Natural Language Processing
- Agents
- Time Series
- Robotics and Control
- Computer Networks and Web Services

The following papers received the Best Paper Award:

1st Place: Semantically Mapping the Web, by *Eduardo Ramirez and Ramon Brena.*
2nd Place: Star Fields: Improvements in Shape-Based Image Retrieval, by *Alberto Chávez-Aragón, Oleg Starostenko and Leticia Flores Pulido*
3rd Place: Keywords Extraction in Clusters of Related Documents, by *Leticia Arco, Damny Magdaleno, Rafael Bello, Manuel Llanes and Libernys Valdés.*

This volume is a result of work of many people. In the first place we thank the authors of the papers included in this volume, for it is the technical excellence of their papers that gives it value. We thank also the members of the International Editorial Board of the volume and the additional reviewers for their hard work on selecting the best papers out of many submissions we received. We would like to thank Yulia Ledeneva, Israel Román and Arturo Téllez, as well as the personnel of the Center for Computing Research of the National Polytechnic Institute, in the first place Oralia del Carmen Pérez Orozco and Ignacio García Araoz, for their indispensable help in the preparation of the volume. The submission, reviewing, and selection process was supported for free by the EasyChair system, www.EasyChair.org.

May, 2007

Sulema Torres
Itzamá López
Hiram Calvo

Table of Contents
Índice

Page/Pág.

Knowledge Acquisition and Neural Networks

Fuzzy Logic

Optimization

Image Processing

Natural Language Processing

Agents

Time Series

Robotics and Control

Computer Networks and Web Services

Knowledge Acquisition and Neural Networks

Knowledge Discovery for Knowledge Based Systems. Some Experimental Results

Rancan C., Kogan A., Pesado P. and García-Martínez R.

Computer Science Doctorate Program. Computer Sc. School. La Plata National University
Software and Knowledge Engineering Center. Postgraduate School. ITBA
Intelligent Systems Laboratory. Engineering School. University of Buenos Aires.
Instituto de Investigación en Informática LIDI, Facultad de Informática, UNLP - CIC Bs As

claudioran@yahoo.com, ppesado@lidi.info.unlp.edu.ar, rgm@itba.edu.ar

Abstract. This paper addresses some considerations based on the state of the involved technologies for the integration of knowledge discovery systems and knowledge based systems centered in automatic knowledge acquisition for experts systems. Some experimental results related to the quality of the generated knowledge bases are shown.

1 Introduction

The knowledge based systems (KBS) or expert systems emulate the human expert behavior in a certain knowledge area. They constitute aid systems to take decisions in different areas such as educational strategic selection [1], environmental variables control [2], neonatology fans configuration [3], agreement in judicial process [4] or the attended generation of activity maps of software development projects [5]. Knowledge based systems to aid decision taking is a one particular knowledge based system.[6]. The knowledge base of an expert system encapsulates in some representation formalism (rules, frames, semantic nets among other), the domain knowledge that should be used by the system to solve a certain problem. The development methodologies of knowledge bases have been consolidated in the last 15 years [7], [8]. The intelligent systems constitute the computer science field which studies and develops algorithms that implement the different learning models and their application to practical problems resolution. Among the problems approached in this field, we can find the one related to knowledge discovering [9]. Knowledge discovery (KD) consists on the search of interesting patterns and important regularities in big information bases [10]. When speaking of knowledge discovery based on intelligent systems or Data/Information Intelligent Mining we refer specifically to the application of machine learning methods or other similar methods, to discover and to enumerate patterns present in this information. One of knowledge discovery paradigms is centered in the knowledge evaluation [11], its structure [12], the distributed acquisition processes [13] and the intelligent systems technologies associated to the knowledge discovery [14]. The interaction between knowledge based systems and discovery systems has antecedents in the paradigm of integrated

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 3-13

Received 09/02/07
Accepted 08/04/07
Final version 20/04/07

architectures of planning and learning based on theories construction [15] and hybrid architectures of learning [16], [17], [18]. In this context, this paper introduces the problem (section 2), an integrative proposal is formulated (section 3), components are identified (section 3.1) and the interaction between them (section 3.2), an example is provided that illustrates partially how the workspace would work (section 4), some experimental results are shown (section 5), finally related work (section 6) future research are adressed (section 7).

2 Problem

Recent works in decision making systems in strategic – operational workspace based on KBS like air control or naval units readiness areas [19] show that it is an open problem to define how KBS can be integrated to knowledge discovery processes based on machine learning that allow them to improve "on-line" the quality of the knowledge base used for decision making. Approaches for solving this type of problem are addressed for incremental improvement of decision making systems in office automation area [20].

3 Toward an Integrative Proposal

In this section the components of the integrative proposal are presented (section 3.1) and the interactions between these components (section 3.2).

3.1 Identification of the Components

3.1.1 The Bases

This section describes: the knowledge base, the concepts dictionary, the examples base, the records base, the clustered records base, the clustered/classification rules base, the discovered rules base and the updated knowledge base.

Knowledge Base. This base contains the problem domain knowledge deduced by the knowledge engineer, which contributes the knowledge pieces (rules) applicable to the resolution of the problem outlined by the user of the system.
Concepts Dictionary. This base stores the registration of all the concepts used in the different knowledge pieces (rules) that integrate the Knowledge Base. For each concept it keeps registration of the corresponding attributes and the possible values of each attribute
Examples Base. This base keeps examples of elements that belong to different classes. The attributes of these examples should keep correlativity or should be coordinated with the attributes of the concepts described in the Concepts Dictionary.
Records Base. This base keeps homogeneous records of information which is associated to some process of knowledge discovery. (I/E clustering).

Clustered Records Base. This base keeps homogeneous records of information which are clustered in classes without labeling (clusters) as a result of applying the clustering process to the Records Base.
Clustering/Classification Rules Base. This base keeps knowledge pieces (rules) discovered automatically as a result of applying the induction process to the Clustered Records Base and the Examples Base
Discovered Rules Base. This base keeps knowledge pieces (rules) related to the problem domain as result of applying the labeling conceptual process to the discovered knowledge pieces (rules) that are stored in the Clustering/Classification Rules Base.
Updated Knowledge Base. This base encapsulates the knowledge that becomes from the integration of the problem domain knowledge pieces (rules) educed by the knowledge engineer and the knowledge pieces (rules) discovered automatically as a result of the application of the processes of clustering/induction to the Records Base or induction to the Examples Base.

3.1.2 The Processes

This section describes the processes: cluster, Inducer, conceptual labeler, knowledge integrator and inference engine.

Cluster. This process is based in the use of self organized maps (SOM) to generate groups of records that are in the Records Base. These groups are stored in the Clustered Records Base.
Inducer. This process is based in the use of induction algorithms to generate clustering rules beginning from the records groups that are in the Clustered Records Base and Classification Rules beginning from the records that are in the Examples Base.
Conceptual Labeler. This process is based on the use of the Concepts Dictionary and the Clustering/Classification Rules Base to generate the Discovered Rules Base. This process transforms the knowledge pieces obtained into pieces of coordinated knowledge with the Knowledge Base.
Knowledge Integrator. This process generates the Updated Knowledge Base from the Discovered Rules Base and the Knowledge Base, solving all the integration problems between them.
Inference Engine. It is the process that automates the reasoning to solve the problem outlined by the user, beginning from the pieces of knowledge available in the Updated Knowledge Base or Knowledge Base.

3.2 Interaction among Components

The interaction among the different components is shown in Figure 1. The Knowledge Base encapsulates the necessary pieces of knowledge (rules) for the resolution of domain problems. This interaction with the inference engine constitutes the Knowledge Based System (Expert System). Beginning from the concepts / attributes / values that are present in the different pieces of knowledge inside the

Knowledge Base, the Concepts Dictionary is built. When a situation of knowledge discovery takes place because the Inducer generated a Clustering/Classification Rules Base, or because this has became from an Examples Base or a Clustered Records Base resultanting of applying the Cluster to a Records Base, the pieces of knowledge (rules) that are in the Clustering/Classification Rules Base can present the characteristic of not being coordinated with the available pieces of knowledge in the Knowledge Base.

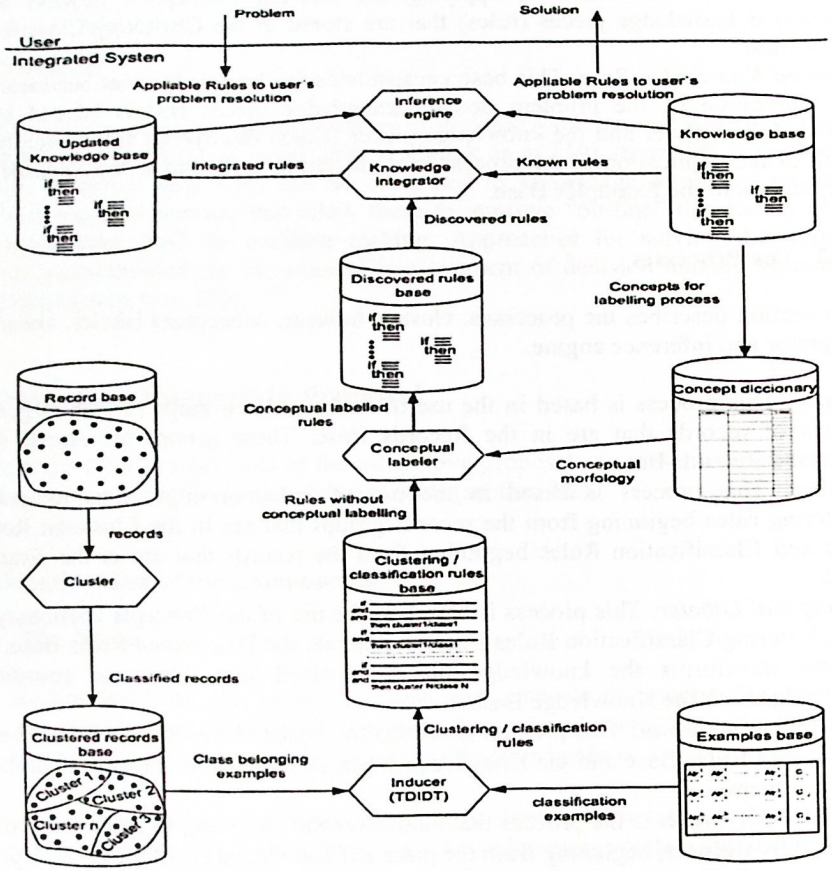


Fig. 1. Interaction among different components

In this context the Conceptual Labeler transforms the knowledge pieces of the Clustering/Classification Rules Base into coordinated knowledge pieces with those rule corresponding to the Knowledge Base generating the Discovered Rules Base. The Knowledge Integrator takes the Discovered Rules Base and (solving the emergent integration problems) integrates it into the Knowledge Base, generating the Updated Knowledge Base, that becomes the new Knowledge Base and the cycle is restarted.

4 An Example in the Ship Operations Cost Domain

Let us consider, for example, the operation costs establishment problem in a ships owner company in function of the ship type to operate in a certain port. Consider the Knowledge Base whose rules are exemplified in table 1. Consider the Concepts Dictionary associated to this Knowledge Base shown in the table 2.

From the Examples Base the Inducer generates the Classification Rules Base shown in the table 4. The Conceptual Labeler identifies the belonging of values to the domain of attributes in Concepts Dictionary generating the Discovered Rules Base shown in the table 5.

The Knowledge Integrator analyzes the Discovered Rules Base, verifying that there are no integration conflicts and proceeds to integrate it to the Knowledge Base generating the Updated Knowledge Base shown in the Table 6. This last one becomes the new Knowledge Base.

Table 1. Knowledge Base

Rules		Rules	
IF	SHIP.SHIP_TYPE= BULK CARRIER	IF	SHIP.SHIP_TYPE= CONTAINER
AND	SHIP.SIZE= LARGE	AND	SHIP.SIZE= LARGE
AND	PORT.PORT_FACILITIES= VERY GOOD	AND	PORT- PORT_FACILITIES= V. GOOD
AND	PORT.ACCESSS= FREEWAY	AND	PORT.ACCESSS:FREEWAY
	THEN COSTS.PIER_LONG= ENLARGE	THEN	COSTS.PIER_LONG= NORMAL
AND	COSTS.MOORING_TIME= HABITUAL	AND	COSTS.MOORING_TIME:SHORT
IF	SHIP.SHIP_TYPE= BULK CARRIER	IF	SHIP.SHIP_TYPE= CONTAINER
AND	SHIP.SIZE= MEDIUM	AND	SHIP. SIZE= MEDIUM
AND	PORT.PORT_FACILITIES= VERY GOOD	AND	PORT.PORT_FACILITIES= VERY GOOD
AND	PORT.ACCESS:FREEWAY	AND	PORT.ACCESSS= FREEWAY
THEN	COSTS.PIER_LONG= ENLARGE	THEN	COSTS.PIER_LONG= NORMAL
AND	COSTS.MOORING_TIME= HABITUAL	AND	COSTS.MOORING_TIME= SHORT
IF	SHIP.SHIP_TYPE= BULK CARRIER	IF	SHIP.SHIP_TYPE= CONTAINER
AND	SHIP.SIZE= SMALL	AND	SHIP.SIZE= SMALL
AND	PORT.PORT_FACILITIES:VERY GOOD	AND	PORT.PORT_FACILITIES= VERY GOOD
AND	ACCESSSS= FREEWAY	AND	PORT.ACCESSS= FREEWAY
THEN	COSTS.PIER_LONG= NORMAL	THEN	COSTS.PIER_LONG= NORMAL
AND	COSTS MOORING_TIME= SHORT	AND	COSTS.MOORING_TIME= SHORT
IF	SHIP.SHIP_TYPE= TANKER	IF	SHIP.SHIP_TYPE= PASENGER
AND	SHIP.SIZE= LARGE	AND	SHIP.SIZE= LARGE
AND	PORT.PORT_FACILITIES= VERY GOOD	AND	PORT.PORT_FACILITIES= VERY GOOD
AND	PORT.ACCESSS:FREEWAY	AND	PORT.ACCESS= FREEWAY
THEN	COSTS.PIER_LONG= NORMAL	THEN	COSTS.PIER_LONG= REDUCED
AND	COSTS.MOORING_TIME:HABITUAL	AND	COSTS.MOORING_TIME= HABITUAL
IF	SHIP.SHIP_TYPE= TANKER	IF	SHIP.SHIP_TYPE= PASENGER
AND	SHIP.SIZE= MEDIUM	AND	SHIP.SIZE= MEDIUM
AND	PORT.PORT_FACILITIES= VERY GOOD	AND	PORT.PORT_FACILITIES= VERY GOOD
AND	PORT.ACCESSS= FREEWAY	AND	PORT.ACCESS= FREEWAY
THEN	COSTS.PIER_LONG= NORMAL	THEN	COSTS.PIER_LONG= REDUCED
AND	COSTS.MOORING_TIME= HABITUAL	AND	COSTS.MOORING_TIME= HABITUAL
IF	SHIP.SHIP_TYPE= TANKER	IF	SHIP.SHIP_TYPE= PASENGER
AND	SHIP.SIZE= SMALL	AND	SHIP.SIZE= SHORT
AND	PORT.PORT_FACILITIES= VERY GOOD	AND	PORT.PORT_FACILITIES= VERY GOOD
AND	PORT.ACCESSS= FREEWAY	AND	PORT.ACCESS= FREEWAY
THEN	COSTS.PIER_LONG= NORMAL	THEN	COSTS.PIER_LONG= NORMAL
AND	COTS.PORT.MOORING_TIME= SHORT	AND	COSTS.MOORING_TIME= SHORT

Table 2. Dictionary of Concepts

Concept	Attribute	Value
SHIP	SHIP_TYPE	BULK CARRIER CONTAINER TANKER PASSENGER
	SIZE	SMALL MEDIUM LARGE
PORT	PORT_FACILITIES	VERY GOOD GOOD REGULAR POOR
	ACCESSS	FREEWAY ROUTE ROAD TRACK
COSTS	PIER_LONG	REDUCED NORMAL ENLARGE
	MOORING_TIME	SHORT HABITUAL EXTEND

Table 3. Examples Base

SHIP_ TYPE	SIZE	PORT_ FAC	ACCESSS	PIER_ LONG	MOORING_ TIME
Bulk Carrier	Large	Very Good	Freeway	Enlarge	Habitual
Bulk Carrier	Medium	Very Good	Freeway	Enlarge	Habitual
Bulk Carrier	Small	Very Good	Freeway	Enlarge	Short
Tanker	Large	Very Good	Freeway	Normal	Habitual
Tanker	Medium	Very Good	Route	Normal	Habitual
Tanker	Small	Very Good	Road	Normal	Short
Container	Large	Very Good	Freeway	Normal	Short
Container	Medium	Very Good	Freeway	Normal	Short
Container	Small	Very Good	Freeway	Normal	Short
Passenger	Large	Very Good	Freeway	Normal	Habitual
Passenger	Medium	Very Good	Freeway	Reduced	Habitual
Passenger	Small	Very Good	Freeway	Reduced	Short

Table 4. Classification Rules Base

Rules	
IF	SHIP_TYPE= CONTAINER
THEN	MOORING_TIME= SHORT
IF	SHIP_TYPE= CONTAINER
THEN:	PIER_LONG= NORMAL
IF	SHIP_TYPE= BULK CARRIER
THEN	PIER_LONG= ENLARGE

Table 5. Discovered Rules Base

Rules	
IF	SHIP SHIP_TYPE= CONTAINER
THEN	COSTS MOORING_TIME= SHORT
IF	SHIP SHIP_TYPE= CONTAINER
THEN	COSTS PIER_LONG= NORMAL
IF	SHIP SHIP_TYPE= BULK CARRIER
THEN	COSTS PIER_LONG= ENLARGE

5 Some Experiments

The improvement of a Knowledge Base with discovered knowledge pieces in automatic way can lead to a degradation of the original Knowledge Base, so it is necessary to explore (theoretically at least) which are the curves of degradation of the quality process of knowledge discovery identifying border conditions for the model in the developed theoretical frame. In order to this a three step experiment which structure is shown in figure 2 has been carry out.

Table 6. Updated Knowledge Base

Rules					
IF	SHIP.SHIP_TYPE= BULK CARRIER	IF	SHIP.SHIP_TYPE= TANKER	IF	SHIP.SHIP_TYPE= PASENGER
AND	SHIP.SIZE= LARGE	AND	SHIP.SIZE= SMALL	AND	SHIP.SIZE= MEDIUM
AND	PORT.PORT_FACILITIES= VERY GOOD	AND	PORT.PORT_FACILITIES= VERY GOOD	AND	PORT.PORT_FACILITIES= VERY GOOD
AND	PORT.ACCESS= FREEWAY	AND	PORT.ACCESS= FREEWAY	AND	PORT.ACCESS= FREEWAY
THEN	COSTS.PIER_LONG= ENLARGE	THEN	COSTS.PIER_LONG= NORMAL	THEN	COSTS.PIER_LONG= REDUCED
AND	COSTS.MOORING_TIME= HABITUAL	AND	COSTS.PORT.MOORING_TIME= SHORT	AND	COSTS.MOORING_TIME= HABITUAL
IF	SHIP.SHIP_TYPE= BULK CARRIER	IF	SHIP.SHIP_TYPE= CONTAINER	IF	SHIP.SHIP_TYPE= PASENGER
AND	SHIP.SIZE= MEDIUM	AND	SHIP.SIZE= LARGE	AND	SHIP.SIZE= SHORT
AND	PORT.PORT_FACILITIES= VERY GOOD	AND	PORT- PORT_FACILITIES= V. GOOD	AND	PORT.PORT_FACILITIES= VERY GOOD
AND	PORT.ACCESS=FREEWAY	AND	PORT.ACCESS= FREEWAY	AND	PORT.ACCESS= FREEWAY
THEN	COSTS.PIER_LONG= ENLARGE	THEN	COSTS.PIER_LONG= NORMAL	THEN	COSTS.PIER_LONG= NORMAL
AND	COSTS.MOORING_TIME= HABITUAL	AND	COSTS.MOORING_TIME= SHORT	AND	COSTS.MOORING_TIME= SHORT
IF	SHIP.SHIP_TYPE= BULK CARRIER	IF	SHIP.SHIP_TYPE= CONTAINER	IF	SHIP.SHIP_TYPE = CONTAINER
AND	SHIP.SIZE= SMALL	AND	SHIP. SIZE= MEDIUM	THEN	COSTS.MOORING_TIME= SHORT
AND	PORT.PORT_FACILITIES=VERY GOOD	AND	PORT.PORT_FACILITIES= VERY GOOD		
AND	ACCESS= FREEWAY	AND	PORT.ACCESS= FREEWAY	IF	SHIP.SHIP_TYPE= CONTAINER
THEN	COSTS.PIER_LONG= NORMAL	THEN	COSTS.PIER_LONG= NORMAL	THEN	COSTS.PIER_LONG= NORMAL
AND	COSTS MOORING_TIME= SHORT	AND	COSTS.MOORING_TIME= SHORT		
				IF	SHIP.SHIP_TYPE= BULK CARRIER
IF	SHIP.SHIP_TYPE= TANKER	IF	SHIP.SHIP_TYPE= CONTAINER	THEN	COSTS.PIER_LONG= ENLARGE
AND	SHIP.SIZE= LARGE	AND	SHIP.SIZE= SMALL		
AND	PORT.PORT_FACILITIES= VERY GOOD	AND	PORT.PORT_FACILITIES= VERY GOOD		
AND	PORT.ACCESS= FREEWAY	AND	PORT.ACCESS= FREEWAY		
THEN	COSTS.PIER_LONG= NORMAL	THEN	COSTS.PIER_LONG= NORMAL		
AND	COSTS.MOORING_TIME= HABITUAL	AND	COSTS.MOORING_TIME= SHORT		
IF	SHIP.SHIP_TYPE= TANKER	IF	SHIP.SHIP_TYPE= PASENGER		
AND	SHIP.SIZE= MEDIUM	AND	SHIP.SIZE= LARGE		
AND	PORT.PORT_FACILITIES= VERY GOOD	AND	PORT.PORT_FACILITIES= VERY GOOD		
AND	PORT.ACCESS= FREEWAY	AND	PORT.ACCESS= FREEWAY		
THEN	COSTS.PIER_LONG= NORMAL	THEN	COSTS.PIER_LONG= REDUCED		
AND	COSTS.MOORING_TIME= HABITUAL	AND	COSTS.MOORING_TIME= HABITUAL		

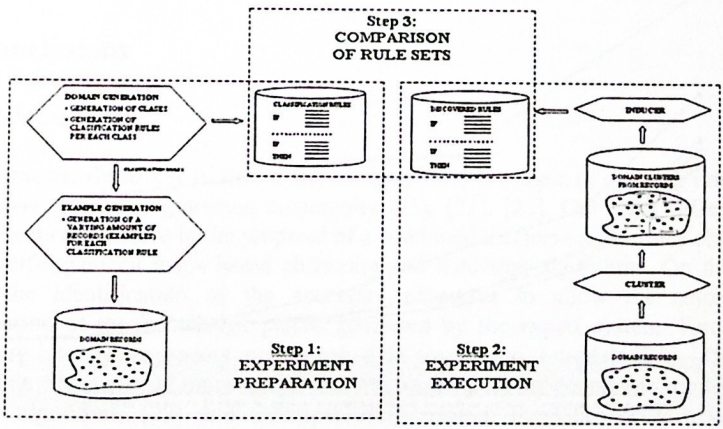


Fig. 2. Structure of he three step experiment

The step 1 consists in experiment preparation. This step involves: [a] domain generation based on: generation of classes and generation of classification rules for each class and [b] examples generation for each classification rule. The output of this step is a classification rules set and a domain records (examples) set. The step 2 consists in experiment execution. This step involves: [a] to apply the cluster process to domain records (examples) set to obtain the domain clusters set and [b] to apply the inducer process to the domain clusters set to obtain the discovered rules set. The step 3 consists on the comparison of the classification rule set from step 1 with the discovered rules set from step 2 the percentage of matching rules defines the experiment success.

5.1 Variables

The experimentation use the following independent variables: [a] *attributes number*: amount of attributes in each classification rule (the same in the examples), [b] *rules per class*: amount of classification rules for determining each domain class, [c] *class possible values*: amount of domain different classes; and the following dependent variable: [b] *rules correctly covered*: percentage of matching rules among classification rules set and discovered rules set.

5.2 Results

The experiments explore the behavior of the processes in domains where classes have associated different amounts of classification rules and the amount of attributes per classification rule can vary and in domains where amount of classes can vary and each class has associated classification rules in which amount of attributes per classification rule can vary. Results of the experiments are shown in figures 3 and 4.

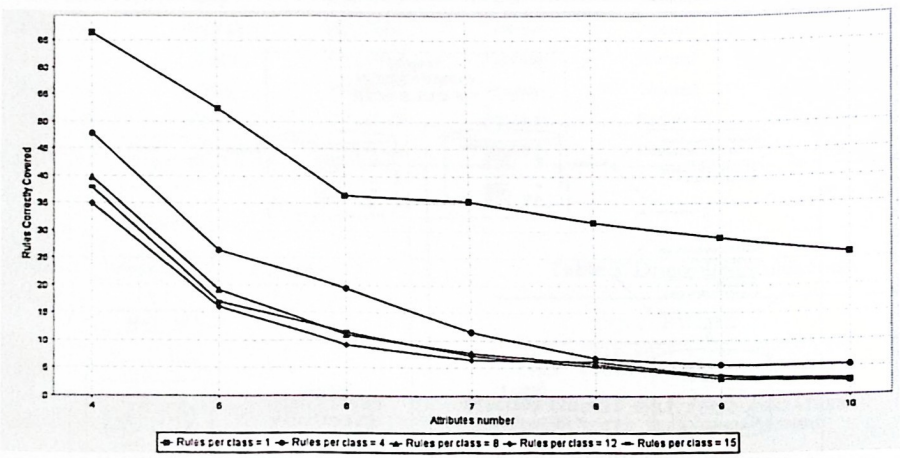


Fig. 3. Domains where classes have associated different amounts of classification rules and the amount of attributes per classification rule can vary.

Figure 1 shows that when domain is complex in terms of amount of attributes needed for classifying (more attributes in a classification rule) or when domain is complex in terms of amount of classification rules needed for identifying a class, the performance (classification rules correctly predicted) of the proposed method (clustering + induction) decreases.

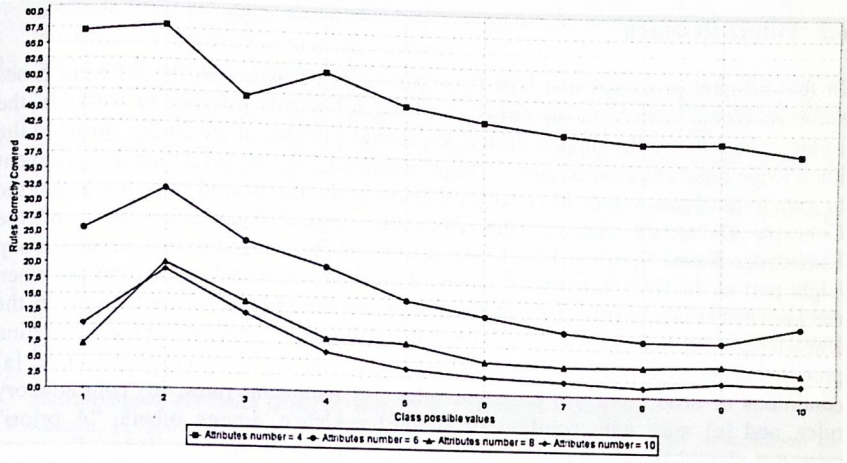


Fig. 4. Domains where amount of classes can vary and each class has associated classification rules in which amount of attributes per classification rule can vary.

Figure 2 shows that when domain is complex in terms of amount of classes the performance (classification rules correctly predicted) of the proposed method (clustering + induction) decreases. Also shows that when the amount of attributes per classification rule for each class decrease, the performance of the proposed method increases.

6 Conclusions

6.1 Related Work

The automatic discovery of useful knowledge pieces is a topic of growing interest in the expert systems engineering community [21], [22], [23]. Our work differs from those mentioned before in the proposal of a combined mechanism for rules obtaining, using self-organized maps based clustering and induction algorithms. On the other hand, the identification of the necessary processes to allow the autonomous assimilation of the knowledge pieces generated by the expert system. Knowledge discovery integration process models based on connectionist models [24], [25], [26], reasoning models based on cases [27], not expected patterns generation models [28], genetic algorithms [29] and technical categorization heuristics [30], have been proposed recently in order to dispose automatic processes for incremental improvement of the intelligent systems response applied to the specific problems

resolution. This proposal differs from the ones mentioned above, in the fact that it proposes a knowledge discovery integration model (rules centered) with expert systems environment, identifying the technology needed to be used to solve this integration.

6.2 Future Research

In the different processes and how these processes interact with the different bases some problems have been identified in whose solution is foreseen to work: In the Inducer: how to use the support groups to provide a degree of credibility (trust) to the knowledge piece (rule) generated. In the Conceptual Labeler: [a] define the treatment to give to attributes values of concepts that are in the discovered rules but not in the Concepts Dictionary that emerges from the original Knowledge Base of the Knowledge Based System and [b] how to rewrite the ownership to a certain group (right part of the rule) in terms of values of attributes of well-known concepts when the knowledge pieces (rules) result from applying the Inducer to the Cluster. In the Knowledge Integrator it should be defined the treatment to apply when the integration process between the rules of the Knowledge Base and the discovered rules arise: [a] conditions of dead point, [b] recurrent rules, [c] redundant rules, [d] contradictory rules, and [e] rules with conflicts of support evidence, among others. "A priori" measures should be developed to establish the quality of the knowledge discovery process and the degree of integrability to the existent Knowledge Base.

References

1. Sierra, E., Hossian, A. y García-Martínez, R. 2003. *Sistemas Expertos que Recomiendan Estrategias de Instrucción. Un Modelo para su Desarrollo*. Revista Latinoamericana de Tecnología Educativa. 1(1): 19-30.
2. Sierra, E., García-Martínez, R., Hossian, A., Britos, P. y Balbuena, E. 2006. *Providing Intelligent User-Adapted Control Strategies in Building Environments*. Research in Computing Science Journal, 19: 235-241
3. Bermejo, F., Britos, P., Rossi, B y García Martínez, R. 2002. *Sistema de Asistencia para la Configuración de Ventiladores OAF en Neonatolgia*. Revista del Instituto Tecnológico de Buenos Aires. 28: 24-68.
4. Gómez, S., Perichinsky, G. y Garcia Martinez, R. 2001. *Un Sistema Experto Legal para la Individualización y Acuerdos para Penas*. Proc. Simposio Argentino de Informática y Derecho. pp. 23-33.
5. Diez, E., Britos, P., Rossi, By García-Martínez, R. 2003. *Generación Asistida del Mapa de Actividades de Proyectos de Desarrollo de Software*. Reportes Técnicos en Ingeniería del Software. 5(1):13-18.
6. García-Martínez, R. y Britos, P. 2004. *Ingeniería de Sistemas Expertos*. Editorial Nueva Librería.
7. Debenham, J. 1990. *Knowledge Systems Design*. Prentice Hall.
8. Debenham, J. 1998. *Knowledge Engineering: Unifying Knowledge Base and Database Design*. Springer-Verlag.
9. Michalski, R. Bratko, I. Kubat, M (eds.) 1998. *Machine Learning and Data Mining, Methods and Applications*, John Wiley & Sons Ltd, West Sussex, England

10. Grossman, R., Kasif, S., Moore, R., Rocke, D. and Ullman, J. 1999. *Data Mining Research: Opportunities and Challenges, A Report of three NSF Workshops on Mining Large, Massive, and Distributed Data*, January 1999, Chicago
11. Jensen D. 2002. *Knowledge Evaluation. Handbook of Data Mining and Knowledge Discovery*. Kloesgen, W. and J. Zytkow (Eds.). Oxford: Oxford University Press
12. Neville J. and D. Jensen 2002. Supporting relational knowledge discovery: Lessons in architecture and algorithm design. Proc. Data Mining Lessons Learned Workshop, 19th International Conference on Machine Learning.
13. Jensen D., Y. Dong, B. Lerner, E. McCall, L. Osterweil, S. Sutton Jr., and A. Wise 1999. *Coordinating agent activities in knowledge discovery processes*. In Proc. International Joint Conference on Work Activities Coordination and Collaboration. pp. 137-146.
14. Britos, P., Hossian, A., García Martínez, R. y Sierra, E. 2005. *Minería de Datos Basada en Sistemas Inteligentes*. 876 páginas. Editorial Nueva Librería. ISBN 987-1104-30-8.
15. García Martínez, R. y Borrajo Millán, D. 2000. *An Integrated Approach of Learning, Planning and Executing*. Journal of Intelligent and Robotic Systems 29(1):47-78.
16. Grosser, H., Britos, P. y García-Martínez, R. 2005. *Detecting Fraud in Mobile Telephony Using Neural Networks*. Lecture Notes in Artificial Intelligence 3533: 613-615.
17. Felgaer, P., Britos, P. and García-Martínez, R. 2006. *Prediction in Health Domain Using Bayesian Network Optimization Based on Induction Learning Techniques*. International Journal of Modern Physics C 17(3): 447-455.
18. Cogliati, M., Britos, P. y García-Martínez, R. (2006). *Patterns in Temporal Series of Meteorological Variables Using SOM & TDIDT*. Springer IFIP Series.
19. Rancán, C. 2004. *Arquitectura de Sistema Híbrido de Evaluación del Alistamiento de Unidades Navales Auxiliares*. Reportes Técnicos en Ingeniería del Software. 6(1): 45-54.
20. Pesado, P. Feierherd G. y Pasini, A. 2005. *Requirement Specifications for Electronic Voting System*. Journal of Computer Science & Technology, 5(4): 312-319 .
21. Hoffmann, F., Baesens, B., Mues, C. and Vanthienen, J. 2006. *Inferring descriptive and approximate fuzzy rules for credit scoring using evolutionary algorithms*. European Journal of Operational Research. (in press).
22. Cao, H., Recknagel, F. Joo, G., Kim, D. 2006. *Discovery of predictive rule sets for chlorophyll-a dynamics in the Nakdong River (Korea) by means of the hybrid evolutionary algorithm HEA*. Ecological Informatics, 1(1): 43-53.
23. Podgorelec, V., Kokol, P., Stiglic, M., Heričko, M., Rozman, I. 2005. *Knowledge discovery with classification rules in a cardiovascular dataset*. Computer Methods and Programs in Biomedicine, 80: S39-S49.
24. Huang, M., Tsou, Y., Lee, S. 2006. *Integrating fuzzy data mining and fuzzy artificial neural networks for discovering implicit knowledge*. Knowledge-Based Systems, 19(6): 396-403.
25. Kasabov, K. 2006. *Adaptation and interaction in dynamical systems: Modelling and rule discovery through evolving connectionist systems*. Applied Soft Computing, 6(3): 307-322.
26. Carpenter, G., Martens, S., Ogas, O. 2005. *Self-organizing information fusion and hierarchical knowledge discovery: a new framework using ARTMAP neural networks*. Neural Networks, 18(3): 287-295.
27. Liu, D., Ke, C. 2006. *Knowledge support for problem-solving in a production process: A hybrid of knowledge discovery and case-based reasoning*. Expert Systems with Applications. (in press).
28. Moreno, M., Quintales, L., García, F., Polo, J. 2004. *Building knowledge discovery-driven models for decision support in project management*. Decision Support Systems, 38(2): 305-317.
29. Kim, M., Han, I. 2003. *The discovery of experts' decision rules from qualitative bankruptcy data using genetic algorithms* Expert Systems with Applications, 25(4): 637-646.
30. Leigh, W., Modani, N., Purvis, R., Roberts, R. 2002. *Stock market trading rule discovery using technical charting heuristics*. Expert Systems with Applications, 23(2): 155-159.

Finding Optimal Neural Network Architecture Using Genetic Algorithms

Fiszelew A., Britos P., Ochoa A., Merlino H.,
Fernández E. and García-Martínez R.

Software & Knowledge Engineering Center. Buenos Aires Institute of Technology.
Intelligent Systems Laboratory. School of Engineering. University of Buenos Aires.

rgm@itba.edu.ar

Abstract. This work deals with methods for finding optimal neural network architectures to learn particular problems. A genetic algorithm is used to discover suitable domain specific architectures; this evolutionary algorithm applies direct codification and uses the error from the trained network as a performance measure to guide the evolution. The network training is accomplished by the back-propagation algorithm; techniques such as training repetition, early stopping and complex regulation are employed to improve the evolutionary process results. The evaluation criteria are based on learning skills and classification accuracy of generated architectures

1 Introduction

The artificial neural networks offer an attractive paradigm for the design and the analysis of adaptive intelligent systems for a wide range of applications in artificial intelligence [1, 2]. Despite the great activity and investigation in this area during last years, that led to the discovery of relevant theoretical and empirical results, the design of neural networks for specific applications under certain designing constrains (for instance, technology) is still a test and error process, depending mainly on previous experience in similar applications [3]. The performance (and cost) of a neural network for particular problems is critically dependant on, among others, the choice of the processing elements (neurons), the net architecture and the learning algorithm [4, 5, 6, 7, 8, 9]. This work is focused in the development of methods for the evolutionary design of architectures for artificial neural networks. Neural networks are usually seen as a method to implement complex non-linear mappings (functions) using simple elementary units interrelated through connections with adaptive weights [10, 11]. We focus in optimizing the structure of connectivity for these networks.

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 15-24

Received 08/02/07
Accepted 08/04/07
Final version 20/04/07

2 Evolutionary Design of Neural Architectures

The key process in the evolutionary approach for topology designing is depicted in figure 1. In the most general case, a genotype can be thought as an array of genes, where every gene takes a value from a properly defined domain [12]. Each genotype codes a phenotype or candidate solution for the domain of interest – in our case a neural architecture class. Such codifications could use genes that take numeric values to represent a few parameters or complex structures of symbols that become into phenotypes (in this case neural networks) by means of a proper decodification process. This process can be extremely simple or quite complex. The resulting neural networks (the phenotypes) can also be equipped with learning algorithms that train them using stimulus from the environment or simply be evaluated in a given task (assuming that the weights of the net are also settled by the coding / decoding mechanism). This evaluation of a phenotype determines the fitness of its corresponding genotype [13, 14]. The evolutionary procedure works in a population of such genotypes, preferably selecting genotypes that code phenotypes with a high fitness, and reproducing them. Genetic operators such as mutation, crossover, etc., are used to introduce variety into the population and to test variants of candidate solutions represented in the current population. In this way, over several generations, the population gradually will evolve toward genotypes that correspond to phenotypes with high fitness. In this work, the genotype only codes the architecture of a neural network with forward connections. The training of the weights for those connections is carried out by the back-propagation algorithm.

3 The Generalization Problem

The topology of a network, that is, the number of nodes and the location and the number of connections among them, has a significant impact in the performance of the network and its generalization skills. The connections density in a neural network determines its ability to store information. If a network doesn't have enough connections among nodes, the training algorithm may never converge; the neural network will not be able to approximate the function. On the other hand, overfitting can happen in a densely connected network. Overfitting is a problem of statistical models where too many parameters are presented. This is a bad situation because instead of learning how to approximate the function presented in the data, the network could simply memorize every training example. The noise in the training data is then memorized as part of the function, often destroying the skills of the network to generalize. Having good generalization as a goal, it is very difficult to realize the best moment to stop the training if we are looking only at the training learning curve. In particular, like we mention previously, it is possible that the network ends up overfitting the training data if the training session is not stopped at the right time. We can identify the beginning of overfitting by using crossed validation: the training examples are split into an training subset and a validation subset. The training subset is used to train the network in the usual way, except for a little modification: the training session is periodically stopped (every a certain number of epochs), and the

network is evaluated with the validation set after each training period. The figure 2 shows the conceptualized forms of two learning curves, one belonging to measures over the training subset and the other over the validation subset. Usually, the model doesn't work so well on the validation subset as it does on the training subset, the design of which the model was based on. The estimation learning curve decreases monotonously to a minimum for a growing number of epochs in the usual way. In contrast, the validation learning curve decreases to a minimum, then it begins to increase while the training continues. When we look at the estimation learning curve it seems that we could improve if we go beyond the minimum point on the validation learning curve. In fact, what the network is learning beyond that point is essentially noise contained in the training set. The early stopping heuristic suggests that the minimum point on the validation learning curve should be used as an approach to stop the training session.

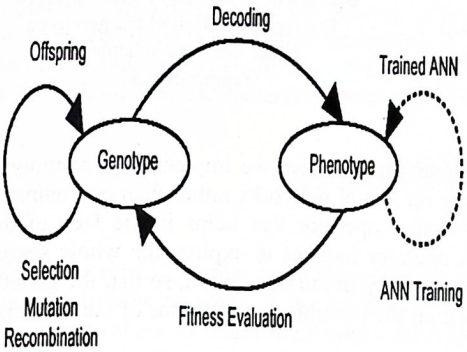


Fig. 1. Design process of evolutionary neural architectures

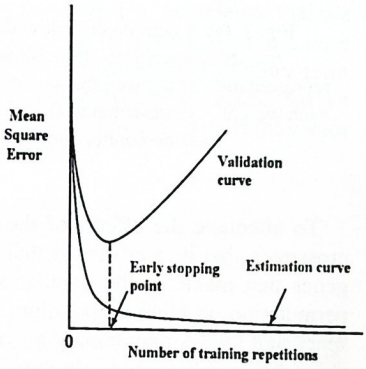


Fig. 2. Representation of the early stopping heuristic based on crossed validation.

The question that arises here is how many times we should let the training subset not improve over the validation subset, before stopping the training session. We define an *early-stopping parameter* β to represent this number of training epochs.

4 The Permutation Problem

A problem that evolutionary neural networks face is the permutation problem. It not only makes evolution less efficient, but also hinders to the recombination operators the production of children with high fitness. The reason is the many-to-one mapping from the coded representation of a neural network to the real neural network decoded,

because two networks that order their hidden nodes in different ways have different representation but can be functionally equivalent, as shown in the figures 3 and 4.

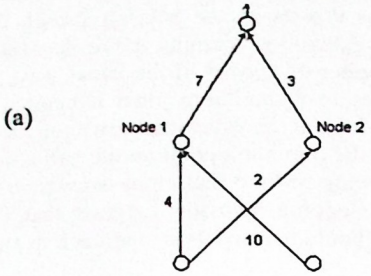


Fig. 3. (a) A neural network with its connection weights; (b) A binary representation of the weights, assuming that each weight is represented with 4 bits. Zero jeans no connection.

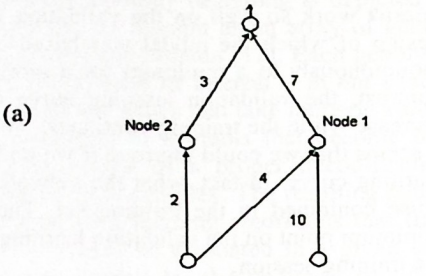


Fig. 4. (a) A neural network that is equivalent to the one in figure 3(a); (b) The representation of the genotype under the same scheme of representation.

To attenuate the effects of the permutation problem, we implement a phenotype crossover, that is, a crossover that works on neural networks rather than on chains of genes that make up the population. Another operator that helps in the face of the permutation problem is mutation. This operator induces to explore the whole search space and allows maintaining a genetic diversity in the population, so that the genetic algorithm is able to find solutions among all the possible permutations of the network.

5 The Noisy Fitness Evaluation Problem

The evaluation of the fitness of the architectures of neural networks will always be noisy if the evolution of the architectures is separated from the training of the weights. The evaluation is noisy because what is used to evaluate the fitness of the phenotype is the real architecture with weights (that is, the phenotype created from the genotype) and the mapping between phenotype and genotype is not one-to-one. Such a noise can deceive to evolution, because the fact that the fitness of a phenotype generated from genotype G1 is higher than the fitness of a phenotype generated from genotype G2 doesn't imply that G1 has truly better quality that G2. To reduce this noise, we train each architecture many times starting from different initial weights chosen randomly. Then we take the best result to estimate the fitness of the phenotype. This method increases the computation time for the fitness evaluation, so a compromise must be achieved among the attenuation of the noise and the number of repetitions for the training.

6 The Complexity-Regularization Problem

As the network design is statistical in nature, we need an appropriate tradeoff between reliability of the training data and goodness of the model. In the context of back-propagation learning, we may realize this tradeoff by minimizing the total risk expressed as:

$$R(w) = \varepsilon_S(W) + \lambda\, \varepsilon_C(w)$$

The first term, $\varepsilon_S(W)$, is the standard *performance measure*, which depends on both the network (model) and the input data. In back-propagation model learning it is typically defined as a mean-square error whose extends over the output neurons of the network and which is carried out for all the training examples on an epoch-by-epoch basis. The second term, $\varepsilon_C(w)$, is the *complexity penalty*, which depends on the network (model) alone; its inclusion imposes on the solution prior knowledge that we may have on the models being considered. We can think of λ as a *regularization parameter*, which represent the relative importance of the complexity-penalty term with respect to the performance-measure term. In the weight-decay procedure that we used, the complexity penalty term is defined as the squared norm of the weight vector w (i.e., all the free parameters) in the network, as shown by:

$$\varepsilon_C(w) = \|w\|^2 = \sum_{i \in C_{total}} w_i^2$$

where the set C_{total} refers to all the synaptic weights in the network. This procedure operates by forcing some of the synaptic weights to take values close to zero, while permitting others to retain their relatively large values. Accordingly, the weights of the network are grouped roughly into two categories: those that have a large influence on the network (model), and those that have little or no influence on it. The weights on the latter category are referred to as excess weights. In the absence of complexity regularization, these weights result in poor generalization by virtue of their high likelihood of taking on completely arbitrary values or causing the network to overfit the data in order to produce a slight reduction in the training error. The use of complexity regularization encourages the excess weights to assume values close to zero, and thereby improve generalization.

7 Experimental Design

The hybrid algorithm that we employ for the automatic generation of neural networks uses a direct coding scheme, and develops the following steps:

1. Create an initial population of individuals (neural networks) with random topologies. Train each individual using the back-propagation algorithm.
2. Select the mother and the father from the population.

3. Recombinate both parents to obtain two children.
4. Mutate each child randomly.
5. Train each child using the back-propagation algorithm.
6. Replace the children into the population.
7. Repeat from step 2 for a given number of generations.

7.1 Parameters Used in the Genetic Algorithm

This algorithm applies a tournament selection (ordinal based) and replacement consists on a steady state update also implemented with a tournament technique. The tournament size is 3. A hundred of generations for the genetic algorithm are carried out in every experiment. All the experiments use a population size of 20. This is a standard value used in genetic algorithms. We make here a compromise among selective pressure and calculation time. The employment of 20 individuals is good to accelerate the development of the experiments without affecting at the results.

7.2 Parameters Used in the Neural Network

Each neural network has 2 hidden layers and is trained over 500 epochs with back-propagation. This value is higher than the one usually used to train neural networks, giving enough time to the training to converge, and so taking advantage of the whole potential of each network. The back-propagation algorithm is based on the sequential training mode; the activation function chosen for each neuron is the hyperbolic tangent. We use a number of back-propagation repetitions equal to 3 to train each neural networks starting from different random initial weights. The best result is then used to estimate the fitness of the network. This algorithm provides an "approximation" to the trajectory in the weight space calculated by the descendant gradient method. The correction Δwji(n) applied to the weight that connects the neuron i to the neuron j is defined by the delta rule:

$$\begin{pmatrix} \text{weight} \\ \text{correction} \\ \Delta w_{ji}(n) \end{pmatrix} = \begin{pmatrix} \text{learning} \\ \text{rate} \\ \eta \end{pmatrix} \cdot \begin{pmatrix} \text{local} \\ \text{gradient} \\ \delta_j(n) \end{pmatrix} \cdot \begin{pmatrix} \text{input signal} \\ \text{of neuron } j \\ y_i(n) \end{pmatrix}$$

A simple way to increment the learning rate and at the same time avoid the risk of instability (oscillations in the net) is to modify the delta rule including a momentum term, lie shown in:

$$\Delta w_{ji}(n) = \alpha \Delta w_{ji}(n-1) + \eta \delta_j(n) y_i(n)$$

where α is usually a positive number called the momentum constant. In the experimentation, the learning rate η is 0.1 and the momentum constant is 0.5.

7.3 The Database Used

The database chosen for the experimentation was taken form a file of datasets in Internet [15]. It consists on data concerning 600 applications for credit cards. Each application represents a sample for the training. The information of the application comprises the input for the neural network during the learning phase. The output is a true/false value that specifies whether the application was accepted or rejected. All the data in the applications is changed into meaningless symbols to protect the confidentiality. The attributes of a sample are similar to:

b,30.83,0,u,g,w,v,1.25,t,t,01,f,g,00202,0,+

In order to present the data to the network, the maximum and minimum values for every attribute into the training set are determined, then they are scaled between -1 and +1. The non-numerical inputs (multiple-choice) are treated in the same way, using discrete intervals.Using these methods of transformation, we obtain a 47 inputs network. The + sign at the end of the example stands for the class of the sample. In this case it will be a + or a -, depending on the approbation of the application, therefore the network has two outputs, one that activates when it is approved and the other when is rejected.

7.4 Cross Validation

The cross validation is employed in the experimentation with the intention of getting better results. The cross validation consists on swapping the training set and the validation set, in the way that each one is used for the opposite purpose. This method assures that any tendency found in the results is, in fact, just tendency, and not causality. Thus, the database is randomly partitioned into two sets of equal size that are in turns used as training and validation subsets.

7.5 Some Results from Experimentation

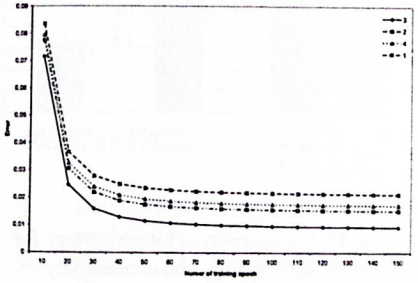


Fig. 5. Complexity-regularization with adjustable regularization parameter: (1) λ=0 (no regularization); (2) λ=0.05; (3) λ=0.01; (4) λ=0.15

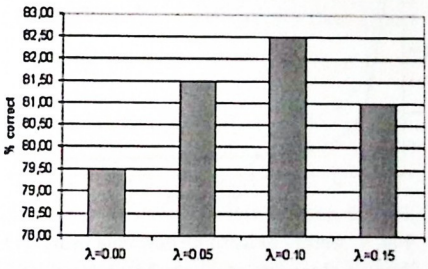


Fig. 6. Comparison of the hit percentage of neural networks generated with different regularization parameters λ.

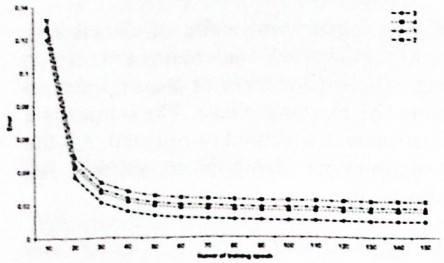


Fig. 7. Early-stopping with adjustable stopping parameter: (1) No early stopping; (2) β=2; (3) β=5; (4) β=10.

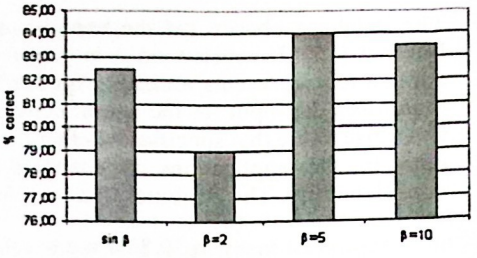


Fig. 8. Comparison of hit percentage of neural networks generated with different early-stopping parameters β.

7.6 Comparison of a Resulting Neural Network with Other Networks

To determine if the evolutionary process is actually improving or not the neural networks concerning with their domain-specific topologies, we compare a resulting net generated by the hybrid algorithm with the best random topology (the one generated in the first generation of the genetic algorithm). These nets are also compared with a topology similar to the one obtained by the hybrid algorithm but 100% connected (or fully connected). We observe the effects of the three different topologies on the convergence of the neural networks while they are trained with a data partition, as depicted in the figure 9. Then we evaluate their classification skills on another data partition, as shown in the figure 10.

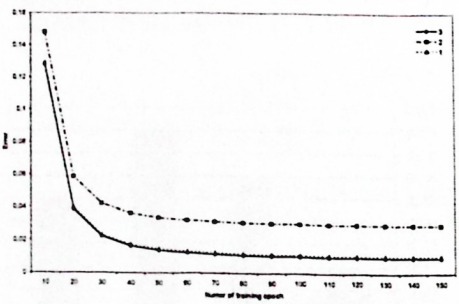


Fig. 9. Ability to learning new data using: (1) Hybrid algorithm topology; (2) Best random topology; (3) Hybrid algorithm topology but 100% connected.

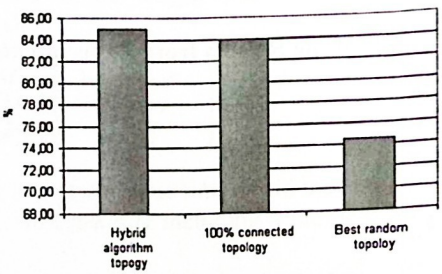


Fig. 10. Comparison of hit percentage for different topology and connectivity.

From figure 9 we see that the neural network generated by the hybrid algorithm is able to learn better a new set of data than the other nets, including the one that implements its same topology but is fully connected. The network generated by the hybrid algorithm also has the best percentage for classifying examples not seen previously, as it is illustrated in figure 10.

8 Conclusions

The real world often has problems that cannot be solved successfully by a single basic technique; each technique has its pros and its cons. The concept of hybrid system in artificial intelligence consists on combining two approaches, in a way that their weaknesses are compensated and their strengths are boosted. The aim of this work is to create a way of generating topologies of neural networks that can easily learn and classify a certain class of data. To achieve this, a genetic algorithm is used to find the best topology that fulfills this task. When the process finishes, the result is a population of domain-specific networks, ready to take new data not seen previously. The analysis of the results of the experiments demonstrates that this implementation is able to create neural networks topologies that in general work better than random or fully connected topologies when they learn and classify new domain-specific data. An aspect that should be examined more deeply is how the cost of a topology should be determined. In the current implementation, the cost is simply the training error of the neural network on a partition of the data set. The question that arises here is if this is the best way to determine the fitness of a topology. Another step to take would be to repeat the experiments for different data sets. Scalability is an important problem in neural networks implementations, therefore it would be interesting to see how the current implementation scales to bigger networks that contain thousands of inputs. A last issue that should be explored is parallelization of the genetic algorithm, especially considering the huge processing times involved during the experimentation. By parallelizing the algorithm, it is possible to increment the population's size, reduce the computational cost and so to improve the performance of the AG. The parallel genetic algorithms or PGAs constitute a recent area of investigation, and very interesting approaches exist such as the Coarse Grained (islands model) PGAs or the Fine Grain PGAs [16].

References

1. Hinton G. E. (1989) *Connectionist Learning Procedures.* Artificial Intelligence, vol. 40, pp. 185-234
2. Hertz J., A. Krogh and R. Palmer (1991) *Introduction to the Theory of Neural Computation.* Reading, MA: Addison-Wesley.
3. Dow R. J. and Sietsma J. (1991) *Creating Artificial Neural Networks that generalize.* Neural Networks, vol. 4, no. 1, pp. 198-209.
4. Haykin Simon (1999) *Neural Networks. A Comprehensive Foundation.* Second Edition. Pretince Hall.

5. Holland J. H. (1975) *Adaptation in Natural and Artificial Systems*. University of Michigan Press (Ann Arbor).
6. Holland, J. H. (1980) *Adaptive algorithms for discovering and using general patterns in growing knowledge-based.* International Journal of Policy Analysis and Information Systems, 4(3), 245-268.
7. Holland, J. H. (1986) *Escaping brittleness: The possibilities of general purpose learning algorithms applied in parallel rule-based systems*. In R. S. Michaiski, J. G. Carbonell, & T. M. Mitchell (Eds.), Machine Learning II (pp. 593-623). Los Altos, CA: Morgan Kaufmann.
8. Holland, J. H., Holyoak, K. J., Nisbett, R. E., & Thagard, P. R. (1987). *Classifier systems, Q-morphisms, and induction*. In L. Davis (Ed.), Genetic algorithms and simulated annealing (pp. 116-128).
9. Honavar V. and L. Uhr. (1993) *Generative Learning Structures and Processes for Generalized Connectionist Networks*. Information Sciences, 70:75--108.
10. Yao Xin (1999) *Evolving Artificial Neural Networks. School of Computer Science*. The University of Birmingham. B15 2TT.
11. Yao X. and Liu Y. (1998) *Toward Designing Artificial Neural Networks by Evolution.* Applied Mathematics and Computation, 91(1): 83-90.
12. Goldberg D. E. (1991) *A comparative analysis of selection schemes used in genetic algorithms*. In Gregory Rawlins, editor. Foundations of Genetic Algorithms, pages 69-93, San Mateo, CA: Morgan Kaufmann Publishers.
13. Rich E. and Knight K. (1991) *Introduction to Artificial Networks*. MacGraw-Hill Publications.
14. Stone M. (1974) *Cross-validatory choice and assessment of statistical predictions*. Journal of the Royal Statistical Society, vol. B36, pp. 111-133.
15. Blake C. L. y Merz C. J. (1998) UCI Repository of machine learning databases. http://www.ics.uci.edu/~mlearn/MLRepository.html Irvine, CA: University of California, Department of Information and Computer Science.
16. Hue Xavier (1997) *Genetic Algorithms for Optimization*. Edinburgh Parallel Computing Centre. The University of Edinburgh.

Mammographic Images Analysis by Use the RMRBF Neural Network

José A. Moreno-Escobar[1], Francisco J. Gallegos-Funes[1], Rene Cruz-Santiago[1] and Volodymyr I. Ponomaryov[2]

Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica
[1]Av. IPN s/n, U.P.A.L.M. SEPI-ESIME, Edif. Z, Acceso 3, Tercer Piso, Col. Lindavista, 07738, Mexico, D. F., Mexico.

[2]Av. Santa Ana 1000, Col. San Francisco Culhuacan, 04430, Mexico, D. F., Mexico.
j.augusto.moreno@gmail.com, fgallegosf@ipn.mx

Abstract. The use of Computer Aided Diagnosis (CAD) for medical analysis is taking relevance in areas as ECG and EEG, but also in cancer detection. In here is proposed the use of the Rank M-Type Radial Basis Function (RMRBF) Neural Network for mammographic images analysis. The proposed neural network uses a proposed RM-estimator in the scheme of Radial Basis Function to train the neural network. To improve the efficiency of the RMRBF, the parameters used to train the network were manipulated in accordance with the RM-estimator theory. From simulation results we observe the classification capabilities of the proposed neural network.

1 Introduction

The artificial neural networks are nonparametric pattern recognition systems that can generalize by learning from examples [1, 4, 0]. They are particularly useful in problems where decision rules are vague and there is no explicit knowledge about the probability density functions governing sample distributions. Therefore, breast cancer detection, in particular mammogram screening, make ideal candidates for application of neural networks [2, 3]. Since the beginning of '90s, different neural networks applications have been considered in breast cancer detection. The neural networks have a potential to improve the performance of computer-based algorithms, especially when used in conjunction with other algorithms. The increase in availability of quality data through publicly accessible databases will provide in the near future more conclusive evidence on utility of neural networks in solving the difficult problem of breast cancer detection. Most frequently, the network architecture of choice in computer-aided mammography is multilayer feed forward, i.e., multilayer perceptron (MP) trained by supervised learning in a form of a backpropagation learning (BPL) law. Typically, neurons are fully connected and employ standard forms of transfer functions. The number of hidden layers is predominantly one, and the number of nodes in the hidden layer is also relatively small. The subject of MP and BPL has been extensively studied; general discussion can be found in text books, and comparative studies and improvements are frequent topics in conferences and journals [10].

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 25-34

Received 10/02/07
Accepted 08/04/07
Final version 18/04/07

In this paper is proposed the Rank M-Type Radial Basis Function (RMRBF) Neural Network for breast cancer detection purposes. The neural network uses a RM-estimator in the scheme of radial basis function to train the neural network according with the schemes found in the references [9, 10]. The use of robust RM-estimators has been introduced for image denoising applications [0, 7, 8]. The combined RM-estimators use different rank estimators such as the median, Wilcoxon and Ansari-Bradley-Siegel-Tukey estimators, and the M-estimator with different influence functions to provide better robustness. The performances of the RM-estimators are better in comparison with original R- and M- estimators [7]. The RMRBF-based training is less biased by the presence of outliers in the training set and was proved an accurate estimation of the implied probabilities.

2 Radial Basis Function Neural Networks

Radial Basis Functions (RBF) have been used in several applications for pattern classification and functional modeling. These functions have been found to have very good functional approximation capabilities [4, 9, 10]. It has been proven that any continuous function can be modeled up to a certain precision by a set of radial basis functions [10]. RBFs have their fundamentals drawn from probability function estimation theory. The structure of the RBF network is depicted in Figure 1. Each network input is assigned to a vector entry and the outputs correspond either to a set of functions to be modeled by the network or to several associated classes.

2.1 Radial Basis Functions Network

Several functions have been tested as activation functions for RBF networks. In pattern classification applications the Gaussian function is preferred, and mixtures of these functions have been considered in various scientific fields.

The Gaussian activation function for RBF networks is given by [9]:

$$\phi_j(\mathbf{X}) = \exp\left[-(\mu_j - \mathbf{X})^T \sum_j^{-1} (\mu_j - \mathbf{X})\right] \tag{1}$$

where $\mathbf{X}$ is the input feature vector, μ_j is the mean vector and Σ_j is the covariance matrix of the jth Gaussian function. Geometrically, μ_j represents the center or location and Σ_j the shape of the basis functions. Statistically, an activation function models a probability density function where μ_j and Σ_j represent the first and second order statistics. A hidden unit function can be represented as a hyper-ellipsoid in the N-dimensional space.

The output layer implements a weighted sum of hidden-unit outputs [4, 9, 10]:

$$\psi_k(\mathbf{X}) = \sum_{j=1}^{L} \lambda_{jk} \phi_j(\mathbf{X}) \tag{2}$$

where L is the number of hidden units, M is the number of outputs with $k=1,...,M$. The weights λ_{kj} show the distribution of the hidden unit j for modeling the output k.

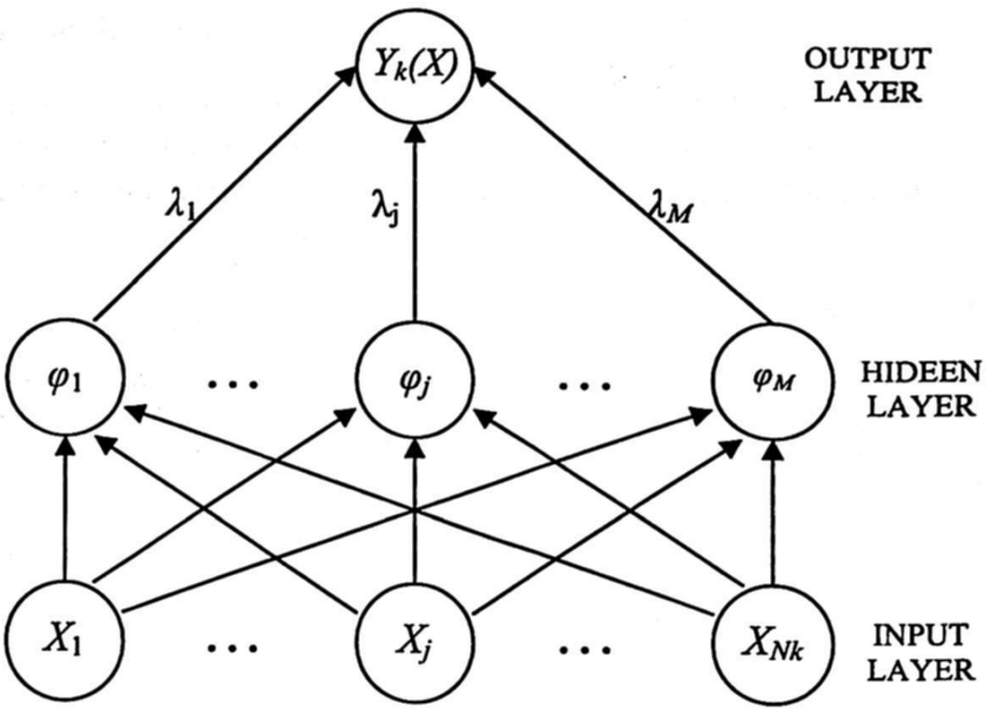


Figure 1. Traditional radial basis function network. Each of N_k components of the input vector X feeds forward to M basis functions whose outputs are linearly combined with weights $\{\lambda_j\}_{j=1}^{M}$ into the network output $Y_k(\mathbf{X})$

2.2 Learning Techniques of RBF Networks

Radial Basis Functions have interesting properties which make them attractive in several applications. A combined unsupervised-supervised learning technique has been used in order to estimate the RBF parameters [9]. In the unsupervised stage, *k*-means clustering algorithm is used to find the pdf's parameters, LMS or instead pseudo-inverse matrix can be used in the supervised stage to calculate the weights coefficients in the neural network [4, 9].

3 Rank M-Type Radial Basis Function Neural Network

In here, we present the use of the RM-estimator as statistic estimation in the Radial Basis Function network architecture. The combined RM-estimators can use different rank estimators such as the median, Wilcoxon or Ansari-Bradley-Siegel-Tukey [0, 7, 8]. The M-estimator uses different influence functions to provide better robustness.

3.1 Activation Function

The Gaussian activation function is the most used function in the RBF networks. In our case we use the inverse multiquadratic function [9]:

$$\phi_j(\mathbf{X}) = \frac{1}{\sqrt{\mathbf{X}^2 + \beta_j^2}} \tag{3}$$

where $\mathbf{X}$ is the input feature vector, β_j is a real constant. In our simulation results β_j=1.

3.2 K-means Algorithm

In our case we used the clustering *k*-means algorithm to estimate the parameters of the RBF neural network [4, 9]. The *k*-means algorithm is used in the unsupervised stage. The input feature vector $\mathbf{X}$ is classified in k different clusters. A new vector $\mathbf{x}$ is assigned to the cluster k whose centroid μ_k is the closest one to the vector. The centroid vector is updated according to,

$$\mu_k = \mu_k + \frac{1}{N_k}(\mathbf{x} - \mu_k) \tag{4}$$

where N_k is the number of vectors already assigned to the *k*-cluster. The centroids can be updated at the end of several iterations or after the test of each new vector. The centroids can be calculated with or without the new vector. By other hand, the steps for the *k*-means algorithm are the following:

Step 1. Select an initial partition with k clusters. Repeat steps 2 through 4 until the cluster membership stabilizes.
Step 2. Generate a new partition by assigning each pattern to its closest cluster center.
Step 3. Compute new cluster centers as the centroids of the clusters.
Step 4. Repeat steps 2 and 3 until an optimum value of the criterion function is found.

3.3 Rank M-type (RM) Estimator

The RM-estimator that is used in the proposal RBF network is the Median M-type (MM) estimator [0, 7]. The non-iterative MM-estimator used as robust statistics estimate of a cluster center is given by,

$$\mu_k = \text{med}\{\mathbf{X}\bar{\psi}(\mathbf{X} - \theta)\} \tag{5}$$

where $\mathbf{X}$ is the input data sample, $\bar{\psi}$ is the normalized influence function ψ: $\psi(\mathbf{X}) = \mathbf{X}\bar{\psi}(\mathbf{X})$, $\theta = \text{med}\{X_k\}$ is the initial estimate, and k=1, 2,...,N_k. The presented estimator is the combined RM-estimator. The *R*-estimator provides good properties of impulsive noise suppression and the *M*-estimator uses different influence functions according to the Huber scheme, providing better robustness. So, it is expected that the performances of combined RM-estimator can be better in comparison with original *R*- and *M*- estimators [7].

3.4 Influence Functions

In our experiments we used the following influence functions [7]:
The simple cut (skipped mean) influence function,

$$\psi_{\text{cut}(r)}(X) = X \cdot 1_{[-r,r]}(X) = \begin{cases} X, & |X| \le r \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

and the Tukey biweight influence function,

$$\psi_{\text{bi}(r)}(X) = \begin{cases} X^2\left(r^2 - X^2\right), & |X| \le r \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

where X is a data sample and r is a real constant. The parameter r depends of the data to process and can be change for different influence functions.

4 Segmentation and Feature Extraction

4.1 Image Collection

To have access to real medical images for experimentation is a very difficult undertaking due to the privacy issues. The data collection that was used in our experiments was taken form many internet sources, but the most important was the MIAS (Mammographic Image Analysis Society) image data base [11, 12, 13]. The MIAS image collection has been used in other studies of automatic mammography classification. Its corpus consists of 322 images, which belong to 3 big categories: normal, benign and malign. There are 208 normal images, 63 benign and 51 malign.

4.2 Segmentation Stage

The first step in image analysis generally consists in a segmentation phase [14]. In this stage the image is divided in regions of interest that contain relevant information for a specific purpose. In our case, due to the irregularity of mammography images, a combination of morphology and threshold methods was used [14, 15, 16]. This way we could divide the mammography in two main regions: a strange object (possible tumor or cancer) and the breast.

4.2.1 Morphology

Morphology techniques offer a powerful method to segment images with irregular shapes or figures as the ones were are using in this work.

The most important morphology operations defined for two sets A and B are [14]:

- Dilation- Consists in growing the geometrical area of a region of interest in an image and can be defined as $A \oplus B$.

- Erosion- Consists in the reduction of the geometrical area of a region of interest in an image, and can be defined as $A \ominus B$.

Combining these techniques the following operations can be implemented [14]:

- Opening, used to eliminate small objects for smoothing a region of interest in an image.

$$A \circ B = (A \ominus B) \oplus B \tag{8}$$

- Closing, also used for smoothing, eliminates small separations or holes of a region of interest in an image. .

$$A \bullet B = (A \oplus B) A \ominus B \tag{9}$$

4.2.2 Thresholding

Thresholding is useful to distinguish pixels that are located in different gray levels (values) and can be considered part of an object. Thresholding values are obtained according to the processed image. In our case, we use two thresholding values: one for the detected object and other one for the breast. The background is not used because it does not offer relevant information, and considering that it is a big part of the whole imageand the processing time is reduced.

4.3 Feature Extraction

There are techniques of geometric measurement that allow the evaluation of some characteristics associated to objects detected in an image. These techniques give us an idea of how compact, bright and smooth an object in an image is [15, 16]. Two of these characteristics are:

Compactness: Is a measure of an object distribution,

$$compactness = \frac{perimeter^2}{area} \tag{10}$$

Contrast: Is the difference between the average gray levels of two objects in an image,

$$contrast = \frac{object\_average - breast\_average}{object\_average + breast\_average} \tag{11}$$

Also, some basic statistics quantities were used:

- Average value (detected object and breast),
- Standard deviation (detected object and breast),
- Range (detected object and breast),

making a total of 8 characteristics used in the proposed RMRBF neural network.

5 Experimental Results

The first step to be done was to select the type of classification that the neural network was going to make. Because all of the literature related focuses only in microcalcification detection, the purpose here was to find also cancer abnormalities. That was the reason to purpose 2 main groups of classification: the first one (Group 1) will be constituted of normal images and benign abnormalities; the second one (Group 2), of images with any kind of microcalficication and malign abnormalities. The classification process is presented in Figure 2.

To train the network for getting the appropriate pdf's parameters were used 32 images (8 normal, 8 benign abnormalities, 8 malign abnormalities, and 4 benign and 4 malign microcalficications), and to probe the efficiency were used 125 images (40 normal, 38 benign abnormalities, 30 malign abnormalities, 8 benign microcalcifications and 9 malign microcalficications), all of them of the MIAS image collection.

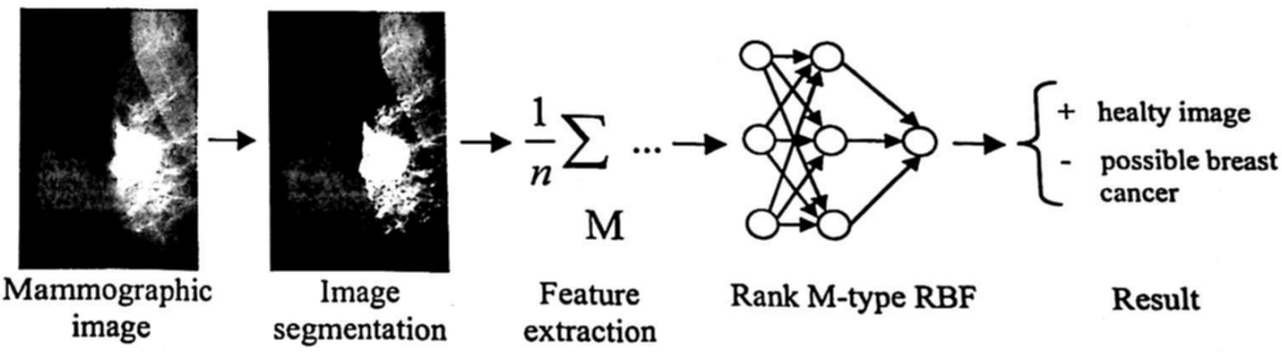


Figure 2. Classification process

The neural network RMRBF propose here is evaluated using the influence functions mentioned in section 3.4, and its performance is compared with the simple RBF algorithm, which was implemented according to its references [9, 10].

Tables 1 to 7 show the experimental results, for being the most important ones, in terms of efficiency, uncertainty and error for the image collection in the case of normal (NORMAL), benign abnormalities (AN_BEN), malign abnormalities (AN_MAL), benign microcalficications (uC_BEN), and malign microcalficications (uC_MAL). Table 1 shows the results obtained using the simple RBF neural network, and Tables 2 to 7 show the results obtained with the proposed RMRBF neural network.

Table 1. Results obtained by the simple RBF algorithm

SIMPLE RBF	NORMAL	AN_BEN	AN_MAL	uC_BEN	uC_MAL	TOTAL
Efficiency	52.50%	47.37%	30.00%	12.50%	55.56%	39.59%
Uncertainty	1.33%	0.00%	0.00%	0.00%	0.00%	0.27%
Error	46.17%	52.63%	70.00%	87.50%	44.44%	60.15%

The described RMRBF neural network with different influence functions has been evaluated with the simple cut and Tukey influence functions. The data that was used to get the pdf's parameters change in accordance with the variation of the r parameter value that was calculated as a factor of the difference between the mean and the minimum value of the data obtained for each characteristic in the training stage. For this reason the maximum value of $r \approx 2$ *(mean-minimum value)*.

Tables 2 to 4 show three relevant results obtained from the proposed RMRBF using Simple Cut influence function with different values of r.

Table 2. Results obtained using the simple function with r=0.3

SIMPLE CUT	NORMAL	AN_BEN	AN_MAL	uC_BEN	uC_MAL	TOTAL
Efficiency	70.00%	68.42%	30.00%	37.50%	66.67%	54.52%
Uncertainty	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Error	30.00%	31.58%	70.00%	62.50%	33.33%	45.48%

Table 3. Results obtained using the simple function with r=0.4

SIMPLE CUT	NORMAL	AN_BEN	AN_MAL	uC_BEN	uC_MAL	TOTAL
Efficiency	72.50%	65.79%	23.33%	62.50%	66.67%	58.16%
Uncertainty	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Error	27.50%	34.21%	76.67%	37.50%	33.33%	41.84%

Table 4. Results obtained using the simple function with r=0.5

SIMPLE CUT	NORMAL	AN_BEN	AN_MAL	uC_BEN	uC_MAL	TOTAL
Efficiency	77.50%	63.15%	36.67%	50.00%	55.56%	56.58%
Uncertainty	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Error	22.50%	36.85%	63.33%	50.00%	44.44%	43.42%

From Tables 2-4 is observed that the results obtained with simple cut RMRBF are better than results shown in Table 1. Also one can see that the change of r value helps us to obtain a higher efficiency.

Tables 5-7 show the results obtained from RMRBF using Tukey influence function with the same values of r used with Simple Cut influence function.

We notice that the best results are given by the Simple Cut influence function that in one result approaches to 60% of efficiency, but there is still a big percentage of error for medical purposes. One important reason might be the irregularity of mammography images that makes difficult the segmentation stage. Examples of proper and improper results can be seen in Figures 3 and 4.

Table 5. Results obtained using the Tukey function with r=0.3

TUKEY	NORMAL	AN_BEN	AN_MAL	uC_BEN	uC_MAL	TOTAL
Efficiency	70.00%	63.16%	30.00%	37.50%	66.67%	53.46%
Uncertainty	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Error	30.00%	36.84%	70.00%	62.50%	33.33%	46.54%

Table 6. Results obtained using the Tukey function with r=0.4

TUKEY	NORMAL	AN_BEN	AN_MAL	uC_BEN	uC_MAL	TOTAL
Efficiency	75.00%	63.16%	26.67%	50.00%	55.56%	54.08%
Uncertainty	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Error	25.00%	36.84%	73.33%	50.00%	44.44%	45.92%

Table 7. Results obtained using the Tukey function with r=0.5

TUKEY	NORMAL	AN_BEN	AN_MAL	uC_BEN	uC_MAL	TOTAL
Efficiency	72.50%	65.79%	36.67%	37.50%	33.33%	49.16%
Uncertainty	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Error	27.50%	34.21%	63.33%	62.50%	66.67%	50.84%

One improvement that could be made to the RMRBF is taking the best efficiency of each one of the 8 characteristics used for all the values of r used in simulations, and combines them in a new RMRBF Neural Network for trying to get better results.

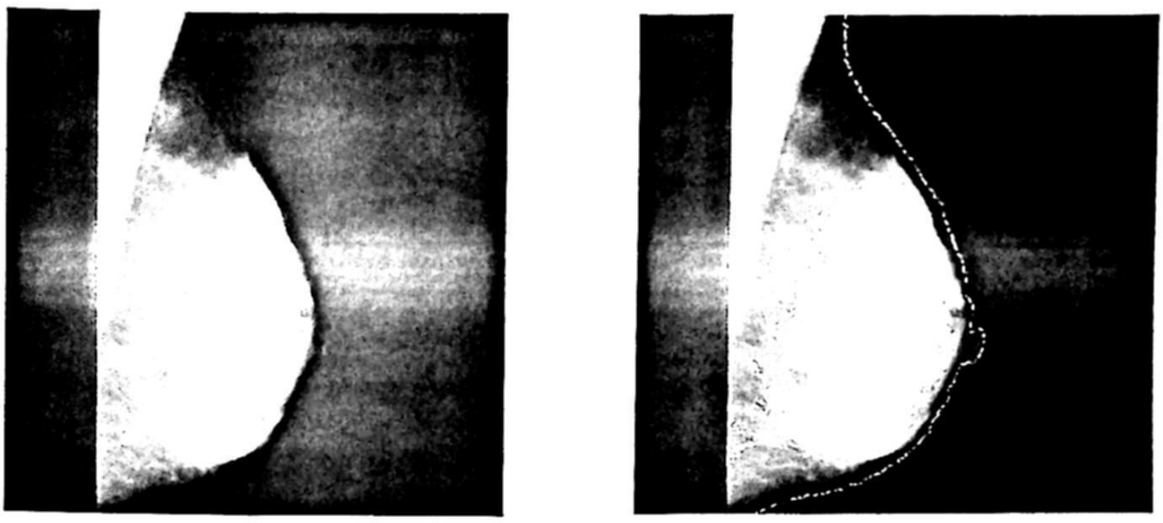


Figure 3. Mammography image with a proper result

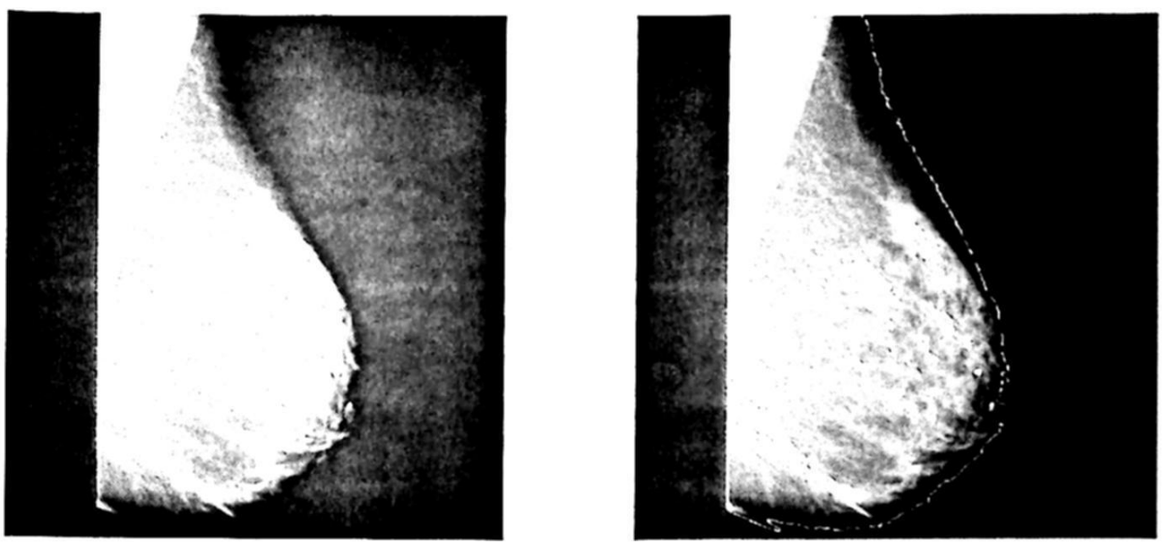


Figure 4. Mammography image with an improper result

6 Conclusions

We present the RMRBF Neural Network, it uses the RM-estimator in the scheme of radial basis function to train the proposed neural network. The results obtained with the use of the proposed RMRBF are better than others results obtained with simple RBF algorithms. Unfortunately the error is still big. The implementation of the Neural Network mentioned at the end of previous section could help, but also another segmentation algorithm should be implemented to see if there is a better differentiation between the regions of interest purposed in this paper.

6.1 Acknowledgements

The authors thank the National Polytechnic Institute of Mexico for its support.

References

1. Hudson D. L., Cohen M. E.: Neural Networks and Artificial Intelligence for Biomedical Engineering. Wiley-IEEE Press, Hoboken, New Jersey (1999)
2. Suri J. S., Rangayyan R. M.: Recent Advances in Breast Imaging, Mammography, and Computer-Aided Diagnosis of Breast Cancer. SPIE Press, Bellingham (2006)
3. Sajda P., Spence C., Pearson J.: Learning contextual relationships in mammograms using a hierarchical pyramid neural network. IEEE Trans. Medical Imag., 21(3) (2002), 239-250
4. Haykin S.: Neural Networks: a Comprehensive Foundation. Prentice Hall, Upper Saddle River, NJ (1994)
5. M. Egmont-Petersen, D. de Ridder, H. Handels, "Image processing with neural networks - a review," Pattern Recognition, vol. 35, (2002) 2279-2301
6. Gallegos-Funes F. J., Ponomaryov V., Sadovnychiy S., Nino-de-Rivera L.: Median M-type K-nearest neighbour (MMKNN) filter to remove impulse noise from corrupted images, IEEE Electronics Letters, 38(15), (2002) 786-787
7. Gallegos-Funes F. J., Ponomaryov V.: Real-time image filtering scheme based on robust estimators in presence of impulsive noise, Real Time Imag., 10(2), (2004) 69-80
8. Gallegos-Funes F. J., Varela-Benitez J. L., Ponomaryov V.: Real-time image processing based on robust linear combinations of order statistics, Proc. SPIE 6063, Real-Time Image Processing 2006, 177-187, San Jose, USA, (2006)
9. Buhmann M. D.: Radial Basis Functions: Theory and Implementations, Cambridge University Press, Cambridge, (2003)
10. Park J., Sandberg J.W.: Universal approximation using radial basis functions network, Neural Computation, vol. 3, (1991) 246-257
11. http://www.wiau.man.ac.uk/services/MIAS/MIAScom.html
12. www.opolanco.es
13. members.tripod.com/~gineco
14. González R. C., Woods R. E.: Tratamiento Digital de Imágenes, Addison Wesley/Díaz de Santos, (1996)
15. Myler H. R., Weeks A. R.: The Pocket Handbook of Image Processing Algorithms in C, Prentice Hall (1993)
16. Ritter G.: Handbook of Computer Vision Algorithms in Image Algebra, CRC Press, Boca Raton-New York (2001)

Fuzzy Logic

Automatic Determination of Parameters for Rule Base Reduction of Complex Fuzzy Control Systems

Yulia Ledeneva,[1] Alexander Gelbukh,[1] Carlos A. Reyes García[2] and René A. García Hernández [3]

[1] Instituto Politécnico Nacional, Centro de Investigación en Computación, México
yledeneva@yahoo.com, gelbukh@gelbukh.com

[2] Instituto Nacional de Astrofísica, Óptica y Electrónica, México
kargaxxi@inaoep.mx

[3] Instituto Tecnológico de Toluca, México
renearnulfo@hotmail.com

Abstract. Fuzzy control methods follow the human way of making control decision. They are based on a usually large number of simple rules describing the reaction of the system under control to each combination of its control variables. Applied systems based on fuzzy control are of great importance in navigation of space vehicles, flight control, missile speed control, industry and manufacture, etc. In many cases, these are complex systems having many variables to control. For such systems, the rule base explodes exponentially in the number of variables. There are methods that considerably reduce the number of rules; however, the performance of such reduced system depends on the choice of some parameters, which have been so far determined only manually based on the experience and knowledge of a skilled system designer. We propose a method that uses a genetic algorithm to automatically determine these parameters for the combination of sensory fusion and hierarchical rule base reduction methods. The implementation process and simulation experiments are presented.

1 Introduction

Since the decade of the 80's, fuzzy logic has been the main source of practical and simple solutions for a great diversity of applications in engineering and science. Fuzzy control algorithms have been the most active area of research in fuzzy logic in the recent years. They are crucial in navigation systems, flight control, satellite control, speed control of missiles, as well as in industrial and manufacturing processes. Some fuzzy control applications to industrial processes produce results superior to those obtained by classical control methods. Moreover, these methods have serious limitations in their expanding to more complex systems, because

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 37-48

Received 03/03/07
Accepted 08/04/07
Final version 22/04/07

currently there is no complete theory of how to predict the behavior of the system after a change in its parameters or variables.

Unlike the classical methods, fuzzy control methodology allows for simple and intuitive expansion to highly complex systems. However, for complex systems the number of fuzzy rules is increased exponentially in the number of variables describing the system: for n variables taking m possible linguistic labels each, m^n rules is needed to construct a complete fuzzy controller. As n grows, the rule base quickly becomes unmanageable. Various methods have been suggested to reduce the number of rules used by the controller.

A combination of sensory fusion and hierarchical methods is very effective in reducing the rule base. However, the popularity of these methods has been limited due to necessity for manually choosing quite a number of parameters: as many as there are variables. So far it has required tedious work and great experience of the system designer in order to find a good—even if not optimal—set of parameters [1].

In this paper, we propose a completely automatic method for choosing the very optimal parameters. Specifically, we use a genetic algorithm (GA) to choose their optimal combination. The paper is organized as follows. Section 2 introduces complex fuzzy control systems. In Section 3, the principles of rule base reduction methods are described. Section 4 proposes the GA-based algorithm that allows for automatically finding the optimal combination of parameters. Experimental results are presented in Section 5 and conclusions in Section 6.

2 Complex Fuzzy Control Systems

A system is complex if its order (the number of control variables) is too high and its model is nonlinear, interconnected with uncertain information flow, so that classical techniques of control theory cannot easily deal with such a system [1]. As the complexity of a system increases, it becomes more difficult and eventually impossible to make a precise statement about its behavior. Fuzzy logic is used in system control and analysis design, because simplifies engineering development; sometimes, in the case of highly complex systems, it is the only way to solve the problem.

The main components of a fuzzy controller are: a process of coding numerical values into fuzzy linguistic labels, an inference engine where the fuzzy rules are implemented, and decoding of the output fuzzy decision variables. Fuzzy control can be implemented by putting the above three stages on a chip or a personal computer.

Dealing with a complex fuzzy system remains a big challenge for any control paradigm to manage the number of the fuzzy IF-THEN rules. When a fuzzy controller is designed for a complex system, often several output and input variables are involved. In addition, each variable is represented by a finite number m of linguistic labels which indicate that the total number of rules is equal to m^n, where n is the number of system variables. As an example, consider $n = 4$ and $m = 5$ than the total number of fuzzy rules will be $k = m^n = 5^4 = 625$. For five variables, $k = 3125$. From the above simple example, it is clear that the application of fuzzy control to any system of significant size would result in a dimensionality explosion.

3 Methods for Reduction of the Rule Base

One of the most important applications of fuzzy set theory has been in the area of fuzzy rule based system. Rule base reduction is an important issue in fuzzy system design, especially for real time Fuzzy Logic Controller (FLC) design. Rule base size can be easily controlled in most fuzzy modeling and identification techniques.

The size of the rule base of complex fuzzy control systems grows exponentially with the number of input variables. Due to that fact, the reduction of the rule base is a very important issue for the design of this kind of controllers. Several rule base reduction methods have been developed to reduce the rule base size. For instance, fuzzy clustering is considered to be one of the important techniques for automatic generation of fuzzy rules from numerical examples. This algorithm maps data points into a given number of clusters [2]. The rule base size can be controlled through the control of the number of cluster centers. However, for control applications, often there is not enough data for a designer to extract a rule base for the controller.

A simple and probably most effective way to reduce the rule base size is to use Sliding Mode Control. The motivation of combining Sliding Mode Control and Fuzzy Logic Control is to reduce the chattering in Sliding Mode Control and enhance robustness in Fuzzy Logic Control. The combination also results in rule base size reduction. However, this approach has its disadvantages as the parameters for the switch function have to be selected by an expert or designed through classical control theory [3].

Anwer [4] proposed a technique for generation and minimization of the number of such rules in case of limited data sets availability. Initial rules for each data pairs are generated and conflicting rules are merged on the basis of their degree of soundness. This technique can be used as an alternative to develop a model when available data may not be sufficient to train the model.

A neuro-fuzzy system [5–9] is a fuzzy system that uses a learning algorithm derived from, or inspired by, neural network theory to determine its parameters (fuzzy sets and fuzzy rules) by processing data samples. Modern neuro-fuzzy systems are usually represented as special multilayer feedforward neural networks (for example, models like ANFIS [8], FuNe [9], Fuzzy RuleNet [10], GARIC [11], HyFis [12] or NEFCON [13] and NEFCLASS [14]). A disadvantage of these approaches is that the determination of the number of processing nodes, the number of layers, and the interconnections among these nodes and layers are still an art and lack systematic procedures.

Jamshidi [1] proposed to use sensory fusion to reduce a rule base size. Sensory fusion combines several inputs into one single input. The rule base size is reduced since the number of inputs is reduced. Also, Jamshidi [1] proposed to use the combination of hierarchical and sensory fusion methods. The disadvantage of the design of hierarchical and sensory fused fuzzy controllers is that much reliance has to be put on the experience of the system designer to establish the needed parameters. To solve this problem, we automatically estimate the parameters for the hierarchical method using GAs.

3.1 Sensory Fusion Method

This method consists in combining variables before providing them to input of the fuzzy controller. These variables are often fused linearly. For example, we want to fuse two input variables y_1 and y_2 (see Figure 1). The fused variable Y will be calculated as $Y = ay_1 + by_2$. Here, it is considered that the input variables of the fuzzy controller are represented by m=5 linguistic labels. So in this case, the number of rules will be thus reduced from 25 to 5. More variables has the fuzzy controller, more reduction can be obtained (see Figure 4).

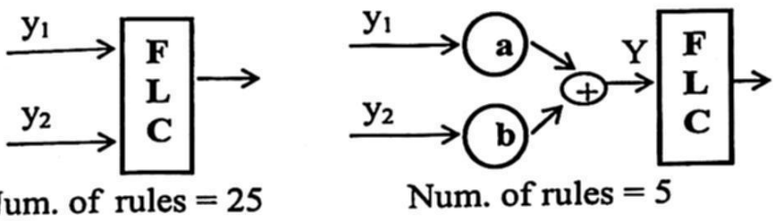


Fig. 1. Fuzzy logic controller's rule base reduction when two variables are fused.

The reduction of the number of rules is optimal if one can fuse all the input variables in only one variable associated. In this case, the number of rules is equal to the definite number of linguistic labels for this variable. But it is obvious that all these variables cannot be fused arbitrarily, any combination of variables has to be reasoned and explained. In practice only two variables are fused: generally the error and the change of error. The fusion can be done through the following rule

$$E = ae + b\Delta e \tag{1}$$

where e and Δe are error and its rate of change, E is the fused variable, and a and b found manually [1].

We want to point out that the manually selection of the parameters a and b convert into fastidious and time-consuming routine. And the described method which permits to reduce significant the number of rules can't be used easily.

3.2 Hierarchical Method

In the hierarchical fuzzy control structure from [1], the first-level rules are those related to the most important variables and are gathered to form the first-level hierarchy. The second most important variables, along with the outputs of the first-level, are chosen as inputs to the second level hierarchy, and so on. Figure 2 shows this hierarchical rule structure.

$$\begin{aligned} &\text{IF } y_1 \text{ is } A_{1i} \text{ and } \ldots \text{ and } y_n \text{ is } A_{ni} \text{ THEN } u_1 \text{ is } B_1 \\ &\text{IF } y_{Ni+1} \text{ is } A_{Ni1} \text{ and } \ldots \text{ and } y_{Ni+nj} \text{ is } A_{Ninj} \text{ THEN } u_i \text{ is } B_i, \end{aligned} \tag{2}$$

where $i,j = 1, \ldots, n$; y_i are the control system's output variables, u_i are the system's control variables, A_{ij} and B_i are linguistic labels; $N_i = \sum_{j=1}^{i-1} n_j \leq n$ and n_j is the number of j-th level system variables used as inputs.

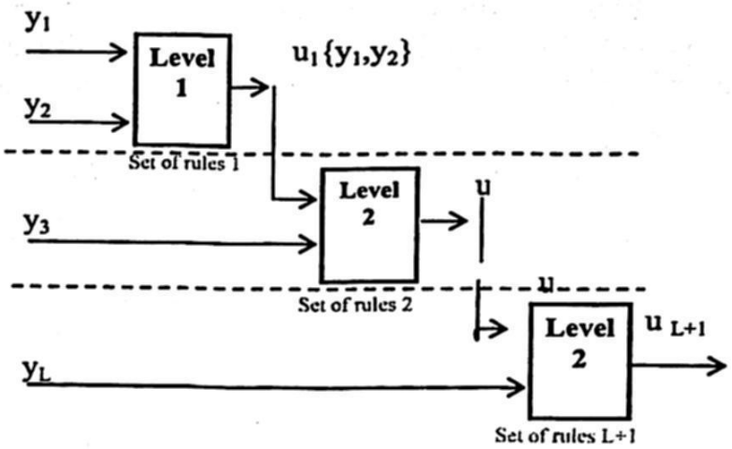


Fig. 2 Schematic representation of a hierarchical fuzzy controller.

The goal of this hierarchical structure is minimize the number of fuzzy rules from exponential to linear function. Such rule base reduction implies that each system variable provides one parameter to the hierarchical scheme. Currently, the selection of such parameters is done manually, which is a tedious and time-consuming.

3.3 Combination of the Methods

The more number of input variables of the fuzzy controller we have, the more it is interesting to combine the methods presented above with a goal to reduce more the rule base. We want to quote, as an example, the combination of the sensory fusion method and the hierarchical method. The sensory fusion method (section 3.1) combined to the hierarchical method (section 3.2) led to an approach illustrated in Figure 3. Initially, the variables are fused linearly, as in Figure 1, and then are organized hierarchically according to a structure similar to that of Figure 2.

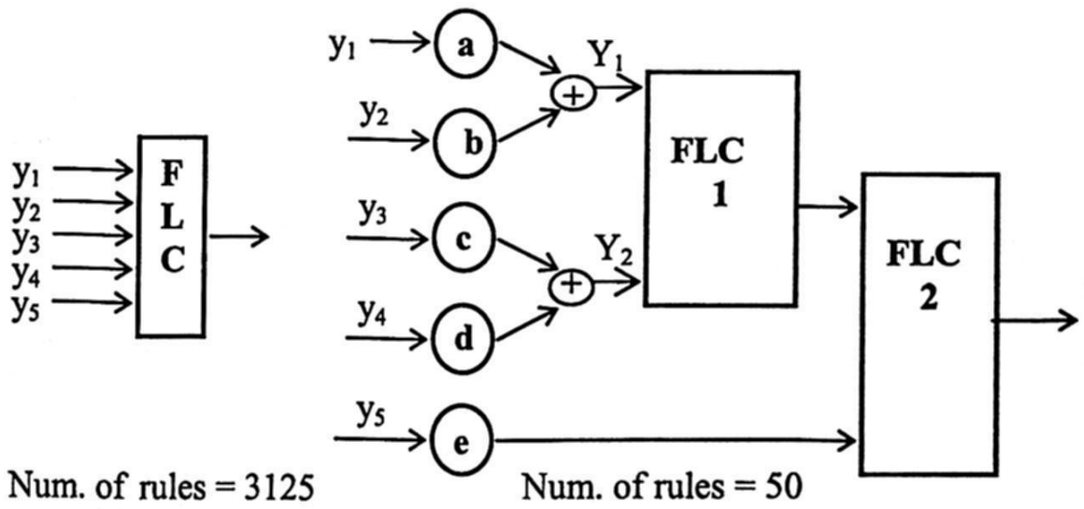


Fig. 3 Combination of the methods for $n=5$ and $m=5$.

The comparison of the sensory fusion method, the hierarchical method and the combination of these rule base reduction methods is presented in Figure 4. Take into

account that the variables are fused here per pair and that on each level of the hierarchy one and only one variable is added. The most significant reduction can be obtained when the sensory fusion and hierarchical methods are combined.

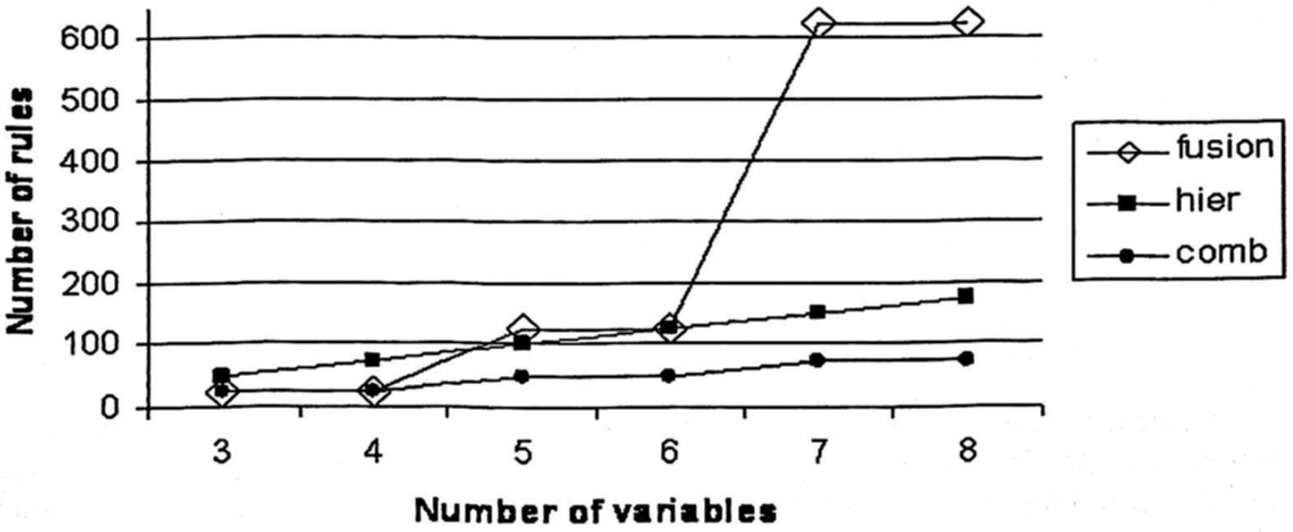


Fig. 4 Comparison of various rule base reduction methods with $m=5$.

4 Genetic Optimization

In this section, we present the proposed method to estimate the parameters of the combination of the sensory fusion and hierarchical rule base reduction methods. The scheme of the proposed method is shown in Figure 5. We have three modules: System Module, Fuzzy Controller Module, and Genetic Algorithm Module. These three modules interconnect in two loops: an internal loop to control a system and an external loop to modify the fusion-hierarchical parameters. The internal loop comprises the fuzzy controller module and the system module. In other words, this loop represents a closed-loop control scheme. The external loop is composed of the genetic algorithm module, the fuzzy controller module, and the system module. The objective of the genetic algorithm module is to estimate the fusion-hierarchical parameters of the fuzzy controller through the minimization of the error between the design specifications and the output of the process. Below we discuss each module of the proposed method.

4.1 Control System Module

The control system is defined as a complex system with p inputs and q outputs:

$$u = [u_1, \ldots, u_p]; \; y = [y_1, \ldots, y_q]; \tag{3}$$

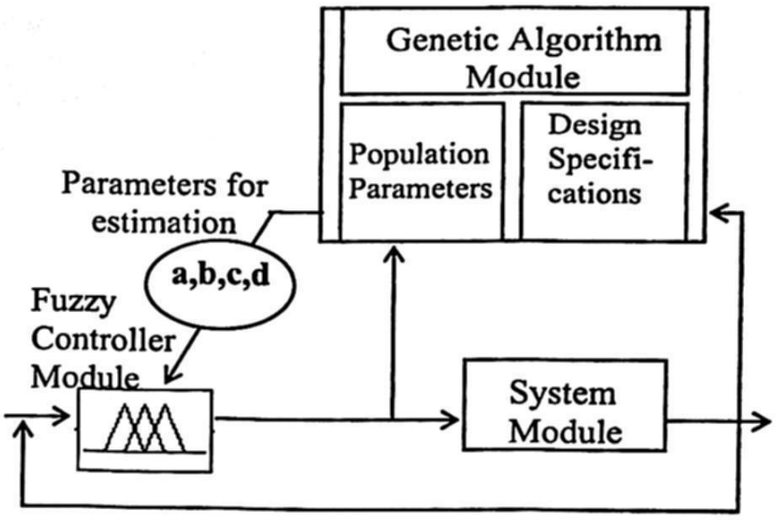


Fig. 5. Scheme of the proposed method.

4.2 Fuzzy Controller Module

The fuzzy controller module is represented by the fuzzy controller of reduced complexity which results after the application of the sensory fusion and hierarchical rule base reduction methods (described above) such that it uses the combination of the fusion-hierarchical parameters.

The fuzzy controller is composed of one or several fuzzy controllers (depending on the number of variables). These controllers are of the Takagi-Sugeno type and each has a maximum of two inputs. The variation of these inputs results from the design of the sensory fusion and hierarchical methods or the output variables of another fuzzy controller.

4.3 Genetic Algorithm Module

The Genetic Algorithm module represents a genetic algorithm that maintains a population of chromosomes where each of which represents a combination of candidate parameters. This genetic algorithm uses data from the system to evaluate the fitness of each parameter in the population. It does this evaluation at each time step by simulating out with each combination of the parameters and forming a fitness function based on the design specifications which characterize the desired performance of the system. Using this fitness evaluation, the genetic algorithm propagates parameters into the next generation via the combination of the genetic operations proposed below. The combination of the parameters that is the fittest one in the population is used in the sensory fusion fuzzy controller.

This allows the proposed method to evolve automatically the combination of parameters from generation to generation (i.e., from one time step to the next, but of course multiple generations could occur between time steps) and hence to tune the

combination of the parameters in response to changes in the system or due to user changes of the specifications in the fitness function of the GA.

We use a standard GA algorithm, which can be summarized as follows:

1. Determine the rule base reduction method and the number of parameters to find.
2. Construct an initial population.
3. Encode each chromosome in the population.
4. Evaluate the fitness value for each chromosome.
5. Reproduce chromosomes according to the fitness value calculated in Step 4.
6. Create offspring and replace the parent chromosomes by the offspring through crossover and mutation.
7. Go to 3 until the maximum number of iterations is met.

4.3.1 Representation

To encode the combination of parameters, chromosomes of length $N \cdot B$ are used, where N is the number of parameters and B the number of bits, which we use to encode the parameters. To decide how many bits to use for each parameter, we should consider the range of all possible values for each of them. For example, suppose that the parameters we want to obtain are positive with one decimal after the dot. To encode all possible values of each parameter we will use 8 bits. In Figure 6, there is one chromosome, representing the combination of parameters, which has $N = 4$ parameters with $B = 8$ bits each. So, the total range of the parameters will be in the interval [0, 256]. To obtain the required precision (one decimal after the dot), we multiply the output values of the parameters by 0.1. As a result, the searching parameters will be in the interval [0, 25.6].

4.3.2 Population

The initial population is randomly generated. Its size is fixed and equal to 50 individuals.

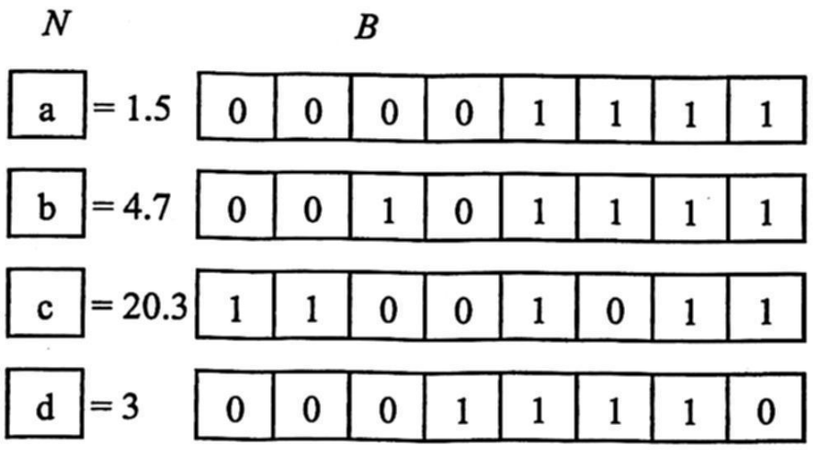


Fig. 6. Example of representation of one chromosome (or one combination of parameters) which has $N = 4$ parameters with $B = 8$ bits each.

4.3.3 Fitness Function

The genetic algorithm maintains a population of chromosomes, each of which represents a different combination of parameters. It also uses a fitness measure that characterizes the closed-loop specifications. Suppose, for instance, that the closed-loop specifications indicate that the user want, for a step input, a (stable) response with a rise-time of t^*_r, a percent overshoot of s^*_p, and a settling time of t^*_s. We propose the fitness function so that it measures how close each individual in the population at time t (i.e., each parameter candidate) is to meet these specifications. Suppose that t_r, s_p, and t_s denote the rise-time, the overshoot, and the settling time, respectively, for a given chromosome (we compute them for a chromosome in the population by performing a simulation of the closed-loop system with the candidate combination of the parameters and a model of the system). Given these values, we propose (for each chromosome and every time step)

$$J = w_1 (t_r - t^*_r)^2 + w_2 (s_p - s^*_p)^2 + w_3 (t_s - t^*_s)^2 \quad (4)$$

where $w_i > 0$, $i = 1, 2, 3$, are positive weighting factors. The function J characterizes how well the candidate combination of the parameters meets the closed-loop specifications; if $J = 0$ it meets the specifications perfectly. The weighting factors can be used to prioritize the importance of meeting the different specifications (e.g., a high value of w_2 relative to the other values indicates that the percent overshoot specification is more important to meet than the others).

Now, we would like to minimize J, but the genetic algorithm is a maximization routine. To minimize J with the genetic algorithm, we propose the fitness function

$$J_{res} = 1/J. \quad (5)$$

We know the design specifications of the system and we can obtain the step response characteristics for each chromosome in the population (rise-time, overshoot, and settling time). If the results given by the GA are in the range of the design specifications of the system, then the fitness function is defined by (4), (5); otherwise, it is set to 1000.

5 Simulation Results

The proposed method was tested in the inverted pendulum control system [15]. The objective of this control system is, on one hand, to maintain the stem of the pendulum in high driving position, on the other hand, to bring the cart towards a given position x_o. The scheme in Figure 7 shows the main components of the system.

The basic variables are:

- the angular position of the stem θ;
- the angular velocity of the stem $\Delta\theta$;
- the horizontal position of cart x;
- the velocity of the cart Δx.

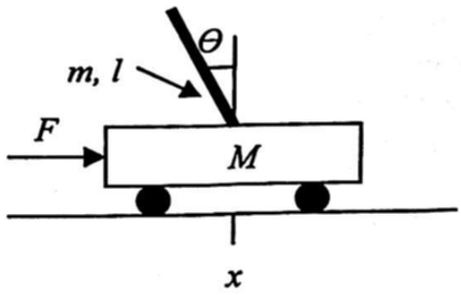


Fig. 7. Inverted pendulum, where $M = 1$ kg is the mass of the cart, $m = 0.1$ kg is the mass of the pendulum, $l = 1$ m is the length to pendulum, F is the force applied to the cart, x is the cart position coordinate, θ is the pendulum angle with vertical.

The design specifications of the inverted pendulum system are:

- the objective position of the cart is 30 cm;
- the overshoot of no more than 5 %;
- the settling time of no more than 5 sec.

The objective position where we must to bring a cart is x_o. The variables to fuse are θ and $\Delta\theta$, e and Δe, where e is the error in position given by $e = x - x_o$ and $\Delta e = \Delta x$. The sensory fusion of the error in position and its variation $X_e = ce + d\Delta e$ combined with a hierarchical method led to the fuzzy controller represented in Figure 8. The first fuzzy controller (FC1) calculates the first control action according to X_e and the angular position θ. In the second fuzzy controller (FC2), it refines the value of preceding control by considering an additional variable $\Delta\theta$. The fuzzy controller based on fusion-hierarchical combination is represented in the Figure 8. The rule bases of FC1 and FC2 are represented in Table 1.

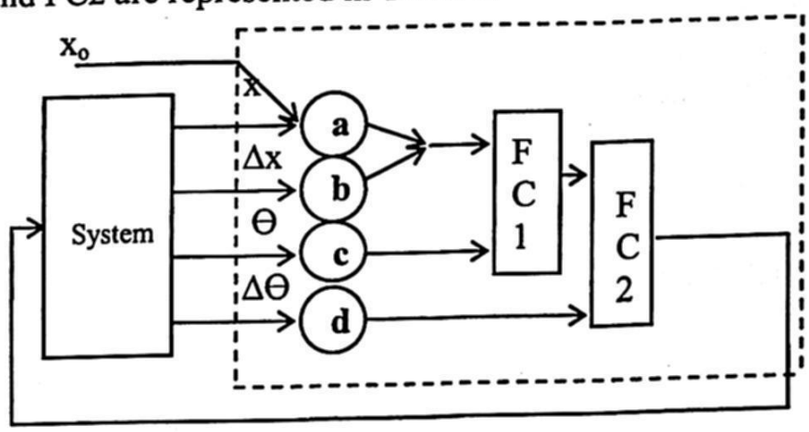


Fig. 8 Fuzzy controller based on the combination of the sensory fusion and hierarchical methods.

The simulation of the inverted pendulum is performed in *Simulink, Matlab* starting from the nonlinear equations [15]. The fuzzy controller is implemented in *Matlab FIS Editor.* The input fuzzy sets are represented by triangular functions (N, Z and P) regularly distributed on the universe of discourse [–1, 1]. The output fuzzy sets are singletons regularly distributed on [–1, 1].

Table 1. Rule bases of the fuzzy controllers FC1 (left) and FC2 (right).

X_e \ $a\theta$	N	Z	P
N		N	
Z	N	Z	P
P		P	

u_1 \ $b\theta$	N	Z	P
N	N	N	Z
Z	N	Z	P
P	Z	P	P

For the reduction with the combination of methods we obtained the following parameters: $a = 25.3$, $b = 10.1$, $c = 3.4$ and $d = 5.5$. With these parameters the horizontal position of the cart is stabilized in 5 seconds with overshoot equal to 0 (see Figure 9), and the behavior of the angle position of the stem of pendulum is shown in the Figure 10.

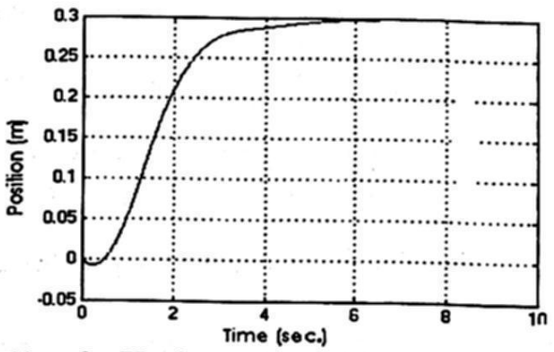


Fig. 9. Horizontal position of the cart.

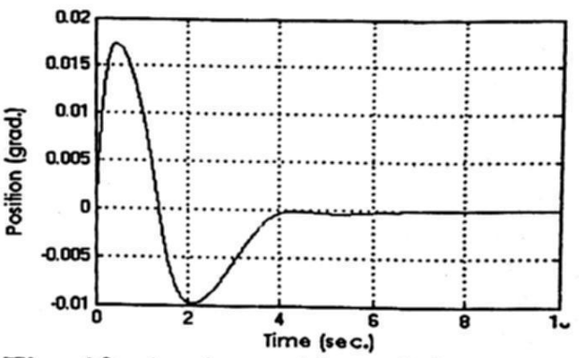


Fig. 10. Angle position of the stem of pendulum.

6 Conclusions

The combination of sensory fusion and hierarchical methods makes it possible to reduce the dimensionality of the control problem more significantly than any of the two methods separately. So far the parameters for such combination have been determined only manually, which requires months of tedious work of highly skilled expert. We suggested a method for automatically finding the optimal combination of such parameters. Our main contribution is the function to be optimized; then we use a genetic algorithm for optimization of this fitness function. We tested the proposed algorithm on a simulation of the inverted pendulum control problem and showed that the fusion-hierarchical parameters for the design specifications of this problem were adequately found.

Due to the fact that the fitness function is based on the design specification of the system, we can apply it to any combination of fusion-hierarchical variables. Another very important advantage is that when the user changes the design specifications, we can obtain the necessary fusion-hierarchical parameters very quickly by using the proposed GA. GA helps not only to automatically estimate the fusion-hierarchical parameters, but also to improve the results obtained by the combination of methods.

References

1. M. Jamshidi. Fuzzy Control Systems. *Springer-Verlag*, chapter of Soft Computing, pp. 42–56, 1997.
2. F. Klawonn, F. Höppner. What is Fuzzy About Fuzzy Clustering? -- Understanding and Improving the Concept of the Fuzzifier. In: M.R. Berthold, at al (Eds.): Advances in Intelligent Data Analysis. *Springer Berlin*, pp. 254-264, 2003.
3. Jonh Y. Hung, et. al. Variable Structure Control: A Survey. *IEEE Trans.on Industrial Electronics,* vol. 40, no.1, pp.2-21, 1993.
4. Zaheeruddin, Anwer M.J. A Simple Technique for Generation and Minimization of Fuzzy Rules. *IEEE International Conference on Fuzzy Systems*, CD-ROM Memories, Nevada, May 2005.
5. Abraham Ajith. Neuro Fyzzy Systems: State-of-the-art. Modeling Techniques. *Connectionist Models of Neurons, Learning Processes, and Artificial Intelligence, Lecture Notes in Computer Science.* Springer-Verlag Germany, Jose Mira and Alberto Prieto (Eds.), Spain, vol. 2084, pp. 269-276, 2001.
6. N. Kasabov, R. Kozma, and W. Duch. Rule Extraction from Linguistic Rule Networks and from Fuzzy Neural Networks: Propositional versus Fuzzy Rules. *Proceedings of the Conference on Neural Networks and Their Applications NEURAP'98*, Marseilles, France, March, 1998, pp. 403-406, 1998.
7. Chia-Feng Juang, Chin-Teng Lin. An On-Line Self-Constructing Neural Fuzzy Inference Network and Its Applications. *IEEE Transaction on Fuzzy Systems,* vol. 6, No.1, pp. 12-32, February 1998.
8. Jyh-Shing Roger Jang, ANFIS: Adaptive-Network-Based Fuzzy Inference Systems, *IEEE Trans. System Man & Cybernetics*, vol. 23, pp. 665-685, 1993.
9. S. K. Halgamuge and M. Glesner. Neural networks in designing fuzzy systems for real world applications. *Fuzzy Sets and Systems*, vol. 65, pp. 1-12, 1994.
10. N. Tschichold-German. RuleNet - A New Knowledge--Based Artificial Neural Network Model with Application Examples in Robotics. PhD thesis, ETH Zerich, 1996.
11. H. R. Berenji and P. Khedkar. Learning and tuning fuzzy logic controllers through reinforcements. *IEEE Trans. Neural Networks*, vol. 3, pp. 724-740, 1992.
12. Kim J., Kasabov N. Hy FIS: adaptive neuro-fuzzy inference systems and their application to nonlinear dynamical systems. Neural Networks, 12, pp. 1301-1319, 1999.
13. D. Nauck, R. Kruse. NEFCON-I: An X-Window Based Simulator for Neural Fuzzy Controllers.*IEEE-ICNN,* WCCI'94 in Orlando, 1994.
14. D. Nauck, R. Kruse. NEFCLASS - A Neuro-Fuzzy Approach for the Classification of Data. *Symposium on Applied Computing,* SAC'95 in Nashville, 1995.
15. Messner William C. and Tilbury Dawn M. Control Tutorials for *Matlab* and *Simulink*: A Web-Based Approach. Addison-Wesley, 1998.

Toward Fuzzy Data Warehouses and Fuzzy Inference

Angélica Urrutia Sepúlveda[1] and Marcela Varas Contreras[2]

[1]Universidad Católica del Maule, Chile
aurrutia@spock.ucm.cl

[2]Universidad de Concepción, Chile
mvaras@udec.cl

Abstract: A Data Warehouse is a data base that stores information for decision making. The characteristics of DWs make the data models and the strategy planning to be different from those used for operational data bases, requiring new techniques and design tools.
This work presents a study on the DW extension for the management of inaccurate information using the fuzzy conjunct theory. Here we display a methodology which incorporates a set of steps to design cube multidimensional components, generating a design star logic, or snow-flake arrangement. To apply, an SQL Server was used and it was extended to apply the FSQL scattered Fuzzy Database Relational engine, as well as incorporating some technical rules of knowledge using SCD, which are an extension of FSQL, to implement inference rules. Finally, we get a Fuzzy Data Warehouse, which implementation, by means of a data base, makes queries on decision making flexible

Key words: Fuzzy Data Warehouse, Fuzzy Data Base, Fuzzy Data Base inference.

1 Introduction

From a beginning, the data bases became fundamental tools for control and handling of commercial operations, reason why in a few years big companies and businesses got a considerable number of information stored in different data sources, reaching considerably higher size. In this way, the data bases have become part of a pillar in all the processes that participate in an organization, integrating different systems that provide quality information for the administration of such organization, which can cause some time problems in accessing, as critical factors in data administration. Figure 1 shows a graph of critical factors and a data base evolution.

A Data Warehousing (DW) is a data base that stores information for the decision making (Peralta 2001; Golfarelli, 1998). This information is built starting from data bases that register the organization's business transactions (operational bases). The DW's objective is to consolidate information from different operational bases making it available to understand the analysis of managerial type of data.

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 49-62

Received 08/03/07
Accepted 08/04/07
Final version 23/04/07

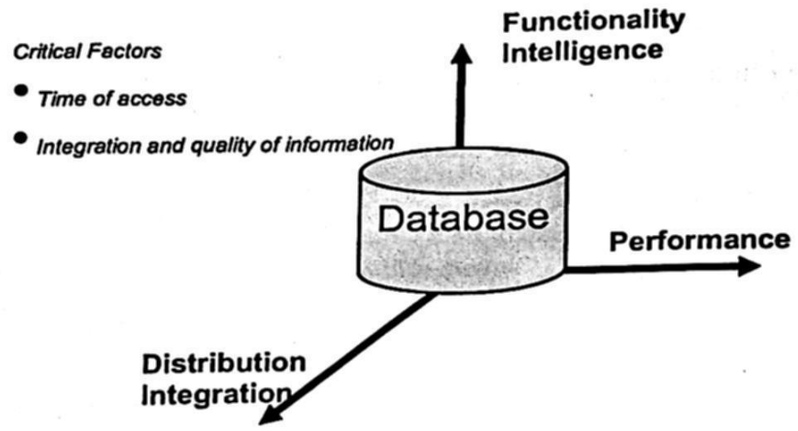


Figure 1: Database Integration and Critical Factors.

The interactive and immediate access to strategic information of an area of business should be treated as a priority. The predominant operations are not the transactions, as in the operational data bases, but the questions that involve great quantity of data and clusters of such transactions, which allow to support the decision making.

The characteristics of the DWs make the design strategies for the operational database not to be generally applicable tohe design of a DW (Kimball, 1996; Inmon, 1996). The data models that represent the data stored in the DW are also different.

If the stored information on a Data Base is considered to be so precise as inaccurate and that the related systems of data bases work with the classicl logic of covering only the type of precise information. Some proposals (Galindo, 1999) allow the use of the inaccurate information by means of Fuzzy Data Bases extending the traditional data bases with the theory of fuzzy groups, allowing the storage of information, treatment and consultation of inaccurate data.

Some jobs in fuzzy DW that are found in both investigations (Kumat et. al, 2005; Feng et. al, 2003) show the typified linguistic labels in some data, which allow to handle the information with extended fuzzy logic. In Server analysis Kumar discloses the data implementing with linguistic labels in SQL Server, while Ling reveals mathematical formality in generating cube's elements with a fuzzy logistics' extension, without reaching application

The investigation presented in this work is the result of the extension from a data warehouse to the inaccurate treatment as well as of its storage, in consultations that allow the managing of information with more flexibility in the decision making. The motors of data used here are SQL Server, FSQL and the inference software SCD, which is an extension of the FSQL. It is necessary to highlight that the implementation was one of the author's degree seminars.

The arranging consists of the theory elements of fuzzy groups, methodology and implementation of a fuzzy data warehouse, conclusions and bibliographical references.

Note: The software used by paper is in Spanish, for what the figures were shown in this language

2 Used Elements

The investigation presented in this document shows the work developed on an SQL Server, where its elements are expanded with the theory of fuzzy groups, types of diffused data and inference treatments using SCD for the DW components.

2.1 Theory of Fuzzy Groups

The concept "fuzzy" comes from the English term "fuzz" that means "confused, blurred, not defined or unfocused, although "fuzzy", in the academic world, is accepted as "ambiguous" and "vague", in the sense of human reasoning. The fuzzy logic, comes from the information that is managed in daily life and which is not always exact or precise, and due to this, it can possess uncertainty and inaccuracy.

A fuzzy group A on a speech universe "n" (finite interval or infinite inside which the fuzzy group can take a value) it is an even group, such as: $A = \{\mu A(x)/x : x \in \Omega, \mu A(x) \in [0,1] \in R\}$ where $\mu A(x)$ it is called the element's degree "x" belonging to the fuzzy group A. This level oscillates between the ends "0" and "1" in the domain of the real numbers:

$\mu A(x) = 0$ indicates that "x" doesn't belong to the fuzzy group A at all.

$\mu A(x) = 1$ indicates that "x" belongs completely to the fuzzy group A.

This definition allows to manage data with a degree of belonging to a group between the extremes "0" and "1", or to generate similar data, understanding the data types from a data base (Zadeh, 1965).

2.2 Linguistic Label

A **linguistic label**, in natural language, is such a word that expresses or identifies a fuzzy set, which may or may not be formally defined. Thus, the fuzzy set A membership function $\mu_A(x)$ expresses the degree in which **x** verifies the category specified by A.

With this definition, we can assure that in our everyday life we use several linguistic labels for expressing abstract concepts such as: "young", "old", "cold", "hot", "cheap", "expensive", etc. The intuitive definition of these labels, not only varies from person to person and depending of the moment, but it also varies with the context in which is applied. For example, a "high" person and a "high" building do not mean the same.

Example 1: If we express the qualitative concept "young" by means of a fuzzy set, where the x axis represents the discourse universe for "age" (in natural whole numbers) and the y axis represents the membership degrees in the interval [0,1]. The fuzzy set that represents such concept could be expressed as follows (considering a discreet universe):

Young = 1/0 + ... + 1/25 + 0.9/26 + 0.8/27 + 0.7/28 + 0.6/29 + 0.5/30 + ... + 0.1/34

The "age" (in total years) would be the discourse universe of "young". The linguistic label "young" would identify this fuzzy set represented by a membership function, if we consider an indiscreet discourse universe from others, such as "adult", "old", etc.

2.3 Types of Fuzzy Data

There are 3 types of attributes susceptible to inaccurate treatment (Galindo, 1999; Urrutia 2003)) and they are classified according to the type of domain underlying them and in that way, allowing to storage vague information or only allowing the imprecise handling of such data:

Type 1: Attribute that are traditional, without vagueness, but they also admit that in their domain it could define some linguistic label to be used in consultations.
Type 2: Attribute that admit data with and without indefinite in form of distribution possibility on an orderly underlying domain. Furthermore, it allows the representation of data type ***Unknown, Undefined and Null.***
Type 3: Attribute that define some labels that are scaled with a defined similar relationship on them, in a way that this relationship shows the criterion by which they look alike on each couple of labels. The same as the previous type, this also allows to represent data labeled as ***Unknown, Undefined and Null.***

2.4 Diffuse Expert System DCS (SCD)

DCS (Diffuse Control System) is a system created at the University of Malaga as a pre-graduate thesis (Escobar, 2003). It is an FSQL extension created by (Galindo, 1999) and it allows to produce inference rules and provides means to carry out data types T1, T2 and T3, with their respective linguistic labels.

It should be clear cut that such an expert system must be fed by an *Expert*, who is a person that has gradually acquired knowledge through a learning and experience period over a specific domain, or having acquired knowledge by using the system.

2.5 Data Warehouse Design

A Data Warehouse (DW) is an administration tool used for the decision making. According to (Inmon, 1996) a DW "is a group of data applied to topics, integrated, volatile and historical, organized to support a process of decision making", from which some transformation stages are considered, like:

1. Drawing out of data: It consists on drawing out data from the DB source and to load it into the DW.
2. Filtering: It consists on filtering unacceptable data into the DW.
3. Format or values modifying: It consists on adapting formats or values so that they comply with the guide lines defined in the DW.
4. Integration: It consists on integrating data coming from two sources.

The most used components in a DW design with Cubes or Hypercubes consisting of Dimensions (with Hierarchies formed by Levels) that are approaches for data analysis, independent Variables and axes in the hypercube, they are also the Measurements, which are values or indicators to analyze, dependent variables, besides variables in the dimensions intersection. On the whole, they are dimensions and measurements form a cube that are represented in a logical design called "shatters" or "snow-flakes", as it may be the case.

A DW has different technological alternatives: ROLAPs: they act directly on BD Relational. MOLAPs: they work on specialized storage. HOLAPs: they try to apply both strategies. Each one of them can be used in the design of a DW.

3 Fuzzy Data Warehouse Methodology and Implementation

Without a doubt (Salas, 2006), the methodology used in the construction of a Data Warehouse is a question of vital importance and it is for that reason that we propose a group of steps to develop a data warehouse that contemplates the extraction of data with the undefined, besides allowing a diffused inference motor. *The development methodology for the construction of the proposed Data Warehouse contemplates 10 stages, which are as follows:*
Situation

1. *Withdrawal and analysis of Requirements:* the necessary sources of data withdrawal are determined from the organization's information system (OLTP), as well as of external sources. Besides that, the user's requirements are established, this is: the Data Warehouse services, restrictions and objectives, obtained through interviews to users.
2. *T-A Transformation:* They are defined as linguistic labels that represent the attributes Type 1 and 2 and, in turn, they produce inputs for the generation of representing the attributes Type 3.
3. *Conceptual Design CMDM:* the conceptual Data Warehouse design is built on this stage, starting from the requirements specified by the users. As it's specified by the CMDM model, cubes are built from dimensions, criterions (standards) and the relationships grouping them.
4. *T-C Transformation*: the Linguistic Label transformations to Cube Dimensions are defined here.
5. *Cube Design:* Defines the activities subjected to analysis and the dimensions that characterize the activities for multidimensional modeling.
6. *T-B Transformation:* Considering the chosen labels from Step 2, the domain stages of each one of the sub-ensembles for attributes Type 1 and Type 2 and the similar relationships for attributes Type 3 that will be used by the inference motor.
7. *Logical design*: defines the outline of ROLAP or MOLAP, producing the logic star design and/or snow flakes and the ETL design (Extraction, Transformation, Load)
8. *Data Warehouse Inference:* defines the rules of knowledge, with acquired information through time and experts.

9. *Implementation*: construction of Data Warehouse Fuzzy and preparation of user's views through Systems Expert.
10. *Representation:* View of a Data Warehouse through a Front End application.

Figure 2 shows a description of the proposed steps for methodology that we have called MFDW (Methodology for Fuzzy Data Warehouse) numbered with each step and their corresponding sequence.

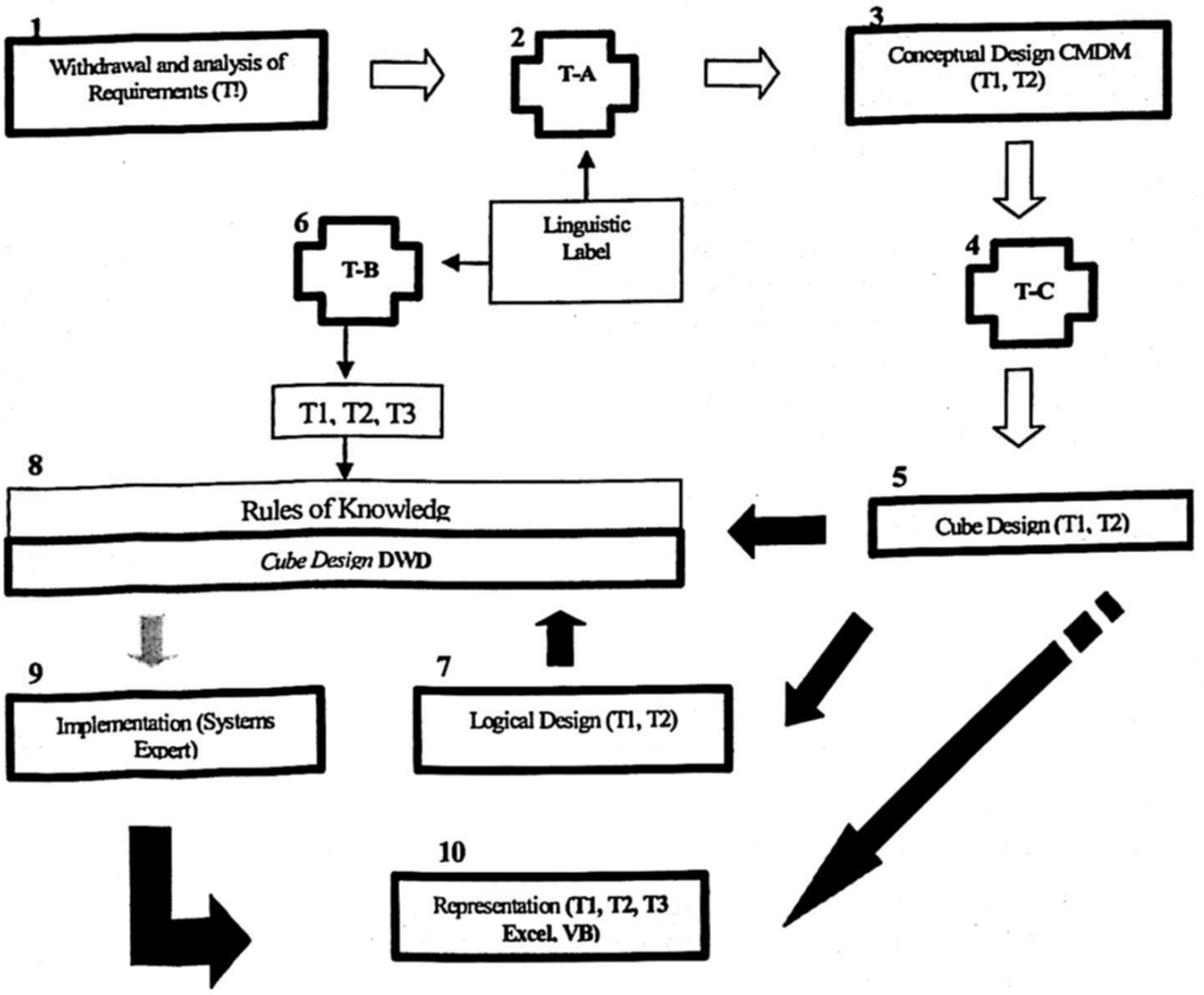


Figure 2: MFDW Methodology for the Development of a Data Warehouse.

Just as mentioned in the section 2.2 the undefined tried with fuzzy logic bears to three types of data T1 (Type 1), T2 (Type 2) and T3 (Type 3), each one of them with a different type of representation. Table 1 displays the stages proposed in the design and implementation of a DW, also, shows the incorporation of inference elements for the administration of information and decision making (Salas, 2006).

Next we show a case where the methodology MFDW is applied and that was part of a degree thesis in Engineering Sciences at the UCM (Universidad Católica Del Maule).

Table 1: Description stages MFDW according to data T1, T2, T3 and inference.

Stages	Type 1 Attributes	Type 2 Attributes	Type 3 Attributes	Inference
analysis of requirements	Receiving Data Reception	Define of data continue	Define of data not continue	None
Conceptual Design	Storing Data	Linguistic Labels	None	Knowledge Rules
Cube's Design	Dimensions, Measurements	Dimensions and Measurements	Dimensions and Measurements	None
Inference Motor	None	None	None	Knowledge Base
Logia Design	Snow Flakes, Star	None	,Snow Flakes, Star	Expert System
Implementation	Excel, Data Report, etc.	Excel, Data Report, Expert System	Expert System	Expert System
Presentation	Normal Data Warehouse	Diffuse or Normal Data Warehouse	Diffuse Data Warehouse	Diffuse Data Warehouse

3.1 Practical Case

The Chilean Telemedicine is in need of periodically getting reports with ECG (Electro-Cardiograms) received data. These reports are requested mainly from the Ministry of Health, which is the main Company client. On the other hand, these statistics are necessary to define the need of present operators at the call center on every schedule and, in this way, plan the shift systems, about which we may say that they have an incidence in the decision making of the Company operations. Also, the statistics throw the quantity of ECGs sent by each client, needed information to collect for the service (billing).

Actually, these reports are carried out through different statistics that are taken based on the ECGs. These statistics are made monthly, checking the received ECGs one by one and analyzing the different points of interest, such as ages, schedules of ECG arrivals, symptoms and histories, among others. An important statistics portion produced in this case allow to collect information for the decision making, being necessary to establish a store for data (DW) to register this information and the storing in time. Some of the data required in this DW have an imprecise behavior, so it is necessary to build a fuzzy DW and, in a special way, some inference rules, mainly for symptoms, so the decision making becomes more efficient.

3.1.1 Collection and Analysis of Requirements

It is the MFDW number one step, if not the most important. They were picked up based on consultation to experts and in the experience on the topic of one of the authors of this work. Not to go into details, in a simplified form, we show the user's requirements according to their *Gender Classification* (Number of ECG received, by sex, for each client), *Symptoms*, *Hour* (Requested ECG received by the time of arrival, number of operators per Schedule in function of demand), *History* (Patient's

history influence on consulting) and *Age* (Number of consulting appointments depending on the patient's age). Some indicators that can be measured are displayed next:

a) Type of diagnosis depending on the patient's type, ECG's timetable, hospital to be sent.
b) Deduce diagnosis based on certain rules of patient's behavior through time.
c) Cardiovascular diagnosis starting from variables, such as hypertension, tobaccoism, etc.

The data base source has the related pattern, which is shown in the figure 3.

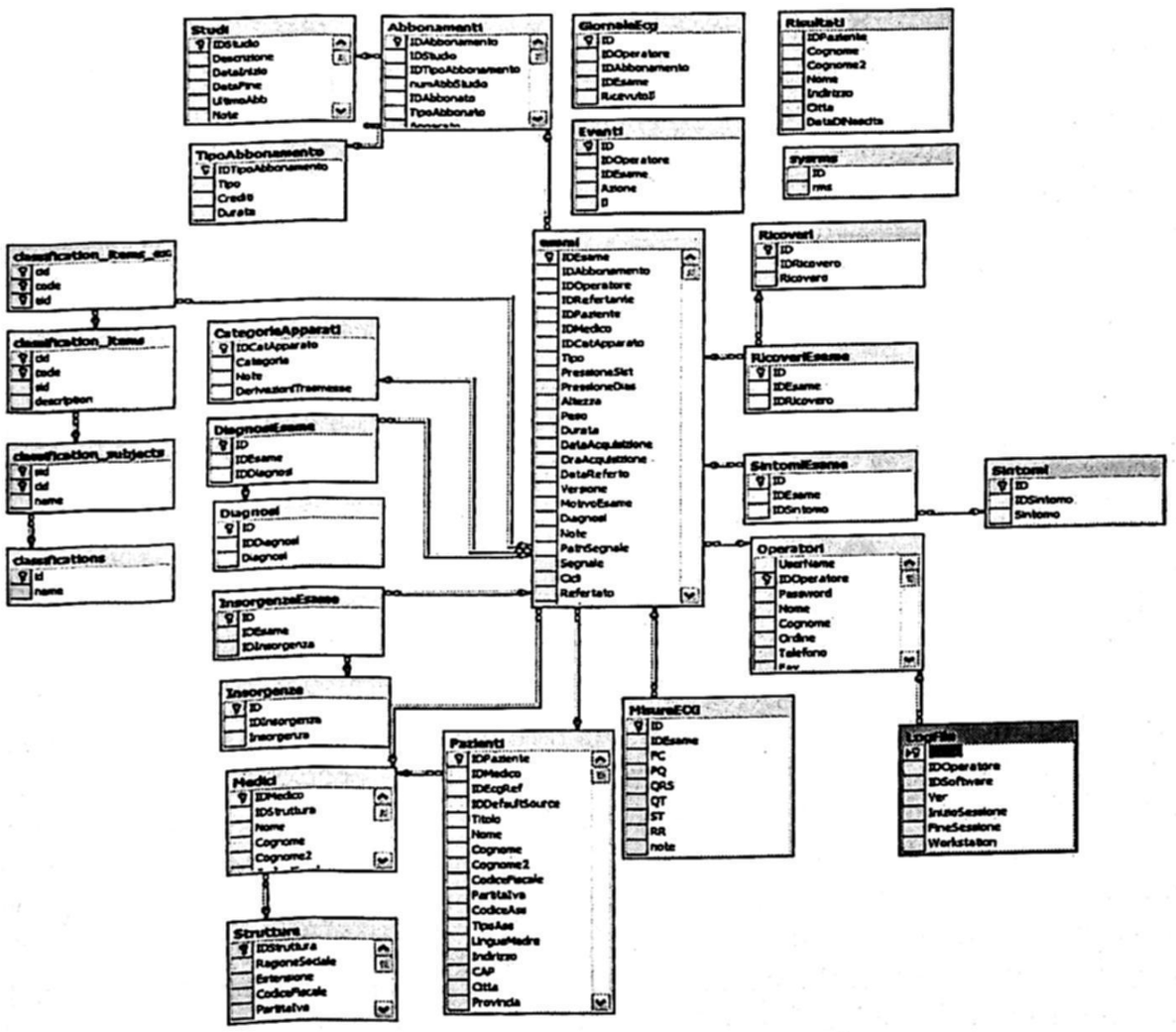


Figure 3: Data Base source (Salas, 2006).

3.1.2 Definition of Linguistic Labels (T-TO)

Under the classification of this Data Base outline the tables "classification_subjects, classification_items, classification_items_exams", were repeated in tables Diagnosis, Symptoms, History and Classification, to manage a descriptive name for such tables. On them, these processes are synchronized via trigger, so that every time an insert is

received in one of the classification tables "classification_subjects, classification_items, classificaton_items_exam", are repeated on of Diagnostic, History and Symptoms tables, which keep the same structure as the tables mentioned previously, but they add a relationship with defined tables to receive fuzzy data, through linguistic labels, which are described next.

STATURE: Linguistic Label: {Under, Normal, High}. Where, the representation is done the same way as in the age label, where the trapezoidal values are:

S(x) = (0, 1.45, 1.65, 2.15).

PHYSICAL STRUCTURE: Linguistics Label: {Small, Average, Big}
Where: Structure Index "Thin" of Corporal Mass is represented as:

P(x) = (0,18,25,40).

AGE: Linguistics Label: {Boy, Young, Adult} where,

L(x)= "1" If the Boy's Age < 8 years, " (8 - Boy's Age)/(10-8)" If the Boy's Age is more than 8 and less than 10 years, " 0" If the Boy's Age is more than 10 years.

L(x)= "0" If the Youth's Age is smaller than 8, or more than 30 years, " (Age of the Youth – 8)/(10-8)" If the Youth's Age is between 8 and 10 years, "1" If the Youth's Age is between 10 and 28 years, "(30 –Youth's Age)/30-28)" If the youth's Age is between 28 and 30 years.

L(x)= "1" If the Adult's Age is 28 years, "(28 - Age Adult)/(100)-30)" If the Adult's Age is more than 30 and less than 100 years, " 0" If the Adult's Age is more than 100 years

Other labels, as time, behave in the same way described previously. Summer - winter - autumn and spring, have their respective months assigned. In the same way "hour", among others.

3.1.3 Conceptual Design.

CMDM will be used (Carpani, 2000) to perform the conceptual design. CMDM distinguishes among dimensions that identify reality objects and relationship dimensions that represent the existent multidimensional relationships among this objects.

- Even: A level representing a group of data
- Dimensions: They represent the approaches for analysis
- Relate Dimensional: They represent cross-links among dimensions.

DIMENSIONS: One of the requirements settles down as a discriminating approach to the diagnosis for the ECGs. This dimension is obtained from the chart of diagnosis. The dimension Diagnosis is composed by a level "*diagnosis*".

The user requests, within the requirements, to know the main symptoms for which an ECG is requested from the patient. The dimension Symptom is formed by a level "*symptoms*". The Patient dimension relates to the patient's data and it is composed by a level "*patients*".

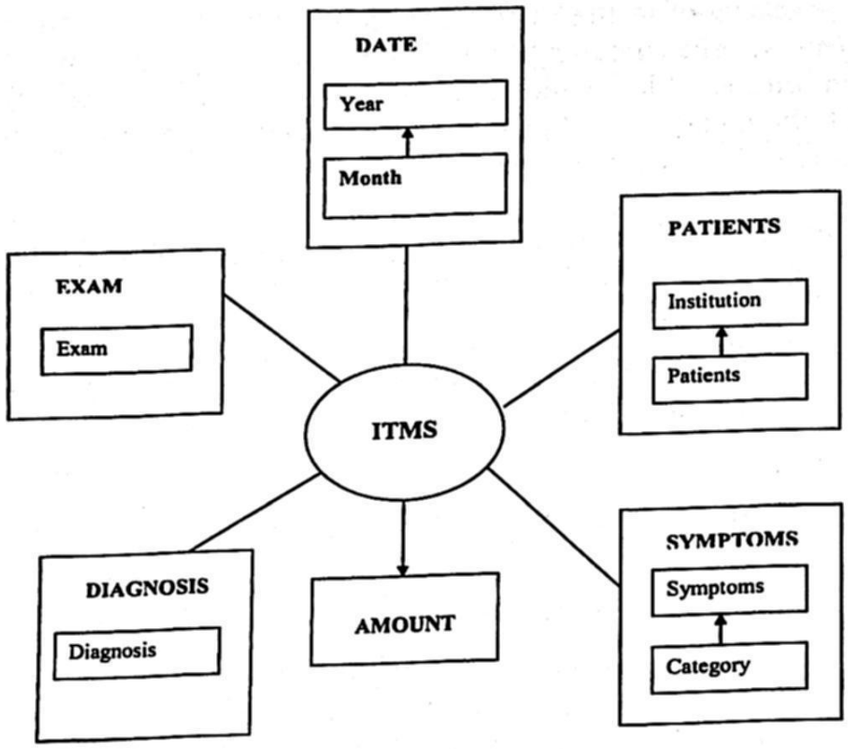


Figure 4: Graphic representation of Dimensional Relational ITMS.

The dimension exam is necessary to obtain the data from the exam. This dimension consists of a level *"exam"*

The dimension "*date*" is one of the requirements for the Data Warehouse development, since most of the data is consulted according to a date or a range of dates. This hierarchy dimension is formed by two levels " *Year and Month*".

The defined dimensions cross-links generated the dimensional relationship "Integrate" (in figure 4 equal ITMS), which links all the dimensions previously defined and from which it could be generated as many cubes as the required measures.

3.1.4 Changing Labels to Dimensions (T-C) and Cube Design.

The MFDW steps 5 and 7 are analyzed in this section, using the relationships shown in figure 4. The Diagnostic cube is believed to represent the measure *Diagnosis Quantity*, which is formed by the Dimensions Diagnosis with the hierarchy Category and the sub-hierarchy Symptoms and History; and the dimension Exam, with the patient's hierarchy.

This cube was implemented in the SQL Analysis Server, just as it's shown in the figure 5, only for accurate data. The cube is built from a star model and a snow flake is made with the diagnosis dimension hierarchies. All this derived from the conceptual model structure change (transformation) in figure 5 as a change to logic design. The implementation for inaccurate data is shown in the section 3.1.5.

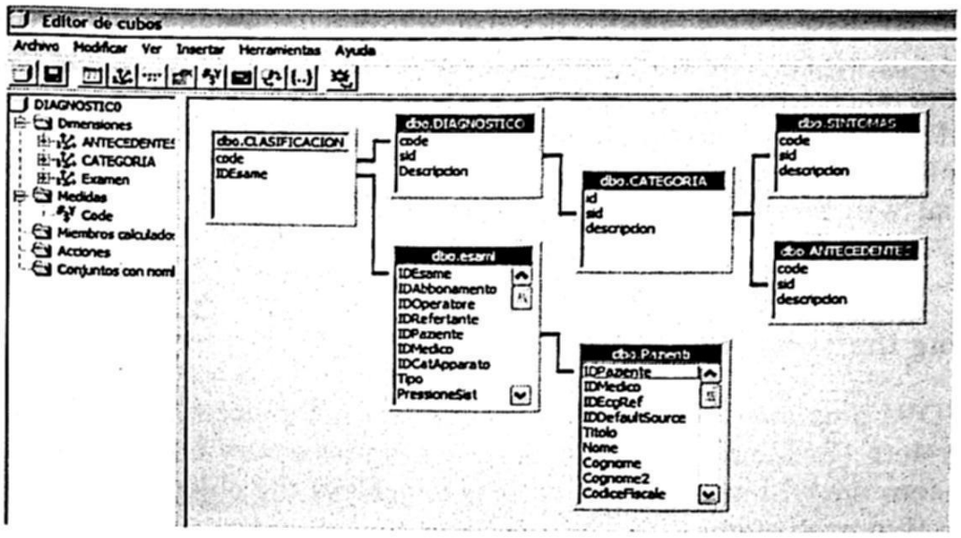


Figure 5: Star Shape Model for Cube Diagnosis.

3.1.5 Representation of a Fuzzy OLAP Cube in a SQL Server

The MFDW step 10 is shown like an extension of the cube generated in the section 3.1.4 using the motor of Galindo's FSQL. To implement FSQL in the SQL Server, which incorporates, as part of the catalog, the relational pattern shown in figure 6, called FIRST: Fuzzy Interface for Relational SysTems (Medina et.al, 1995: Galindo et. al, 2005) in Galindo's FSQL (Galindo, 1999).

Data-Base Implementing in FIRST and the FMB: The Data Base implementing in FIRST (figure 6) in the Data Base SQL Server allows to incorporate the way how the fuzzy and classic attributes are represented in a BDRD.

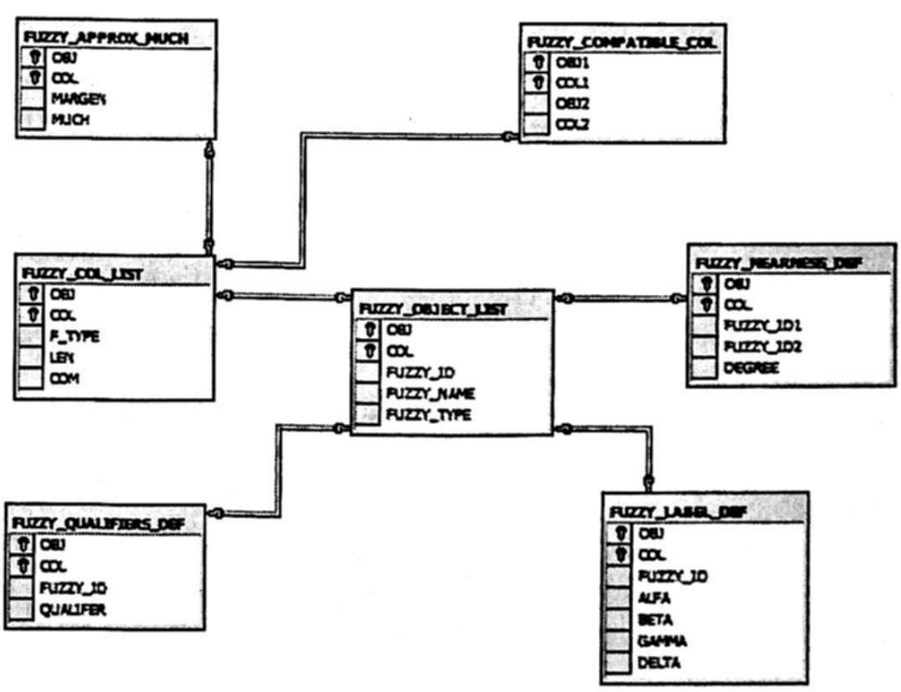


Figure 6: Tables in FIRST that incorporates to the SQL Server in Catalog.

Each one of these tables is filled according to the definition of the linguistic labels for data T1, T2 or T3. The filling of each one of them is discussed next (Salas, 2006) with the data taken from the Data Base sources.

The tables will be filled with the data of the labels shown in the section 3.1.2. With these definitions you are under conditions to implement the BDRD. On this respect, the data that will be stored in FIRST is the one that has been shown in the previous points.

3.1.6 Using the Fuzzy Control System

In the MFDW data steps 9 and 10 we will use a computer system called of Diffuse Control System (Escobar, 2003), DCS, to get the necessary inference for the system. To this system we will make modifications to achieve the objective, which takes us to develop an Expert System.

Such an Expert System should be fed by an expert who can be a person whose knowledge has been acquired gradually through a period of learning and experience in a specific field, or, fed by a system that offers him knowledge through its use (practice). In the application DCS, the attributes Type 2 and Type 3 will be implemented. For Type 2 we will use the symptoms and a patient's history, while in Type 3 we will use diagnosis (show section 2.3). The knowledge rules are obtained through the Data Warehouse.

To the traditional SQL language are added: the linguistic labels, the fuzzy comparators, the logical connectors, the execution threshold and the fuzzy constants (Galindo, 1999). In the application DCS to be used, this SQL extension is found in the code source, where each one of the comparators, connectors, etc., are defined (Salas, 2006)

Next, the development and the use of this system are shown. The first input data input was to create a Project applying DCS, where the user will be *Seminar*; the title, *Cardiology* and as observation, we will define ***inference* of *Cardiovascular Diagnosis.***

Entrance Variables. We will use the Entrance Variable *"Hypertension"*, called by their acronym HTA and *Tobaccoism*, as history of a patient. Both with fuzzy grade of incidence, Low, Average, and High. According to Step 2 of the MFDW pattern proposed in Step 3.1.2 the defined linguistic labels are now implemented.

Output Variables. As Output Variables we will use the possible Diagnosis, according to the inputs in this case. Our possible output variables will be the category *Necrosis* and inside those we will find *Isquemia*. The attributes type 3, *Diagnosis,* are noticed as **output attributes.**

We will create the rules of knowledge showing high HTA with High Tobacconism, getting Necrosis, corresponding to Isquemia Anteroseptal. Figure 7 A) shows a picture of this stage.

Inference of Diagnosis. The inference process of diagnostic is carried out through acquired knowledge providing diagnostic probabilities according to the input variables and to the rules inserted in the inference motor. Figure 7 B) shows a picture of this stage.

The system shown here was of great advantage to the final user, allowing to have pre-defined diagnosis and, over all, providing information more related to the human

treatment, such as uncertain data. A good portion of the grades of domain and the created linguistic labels were obtained from the System's Expert (Escobar, 2003).

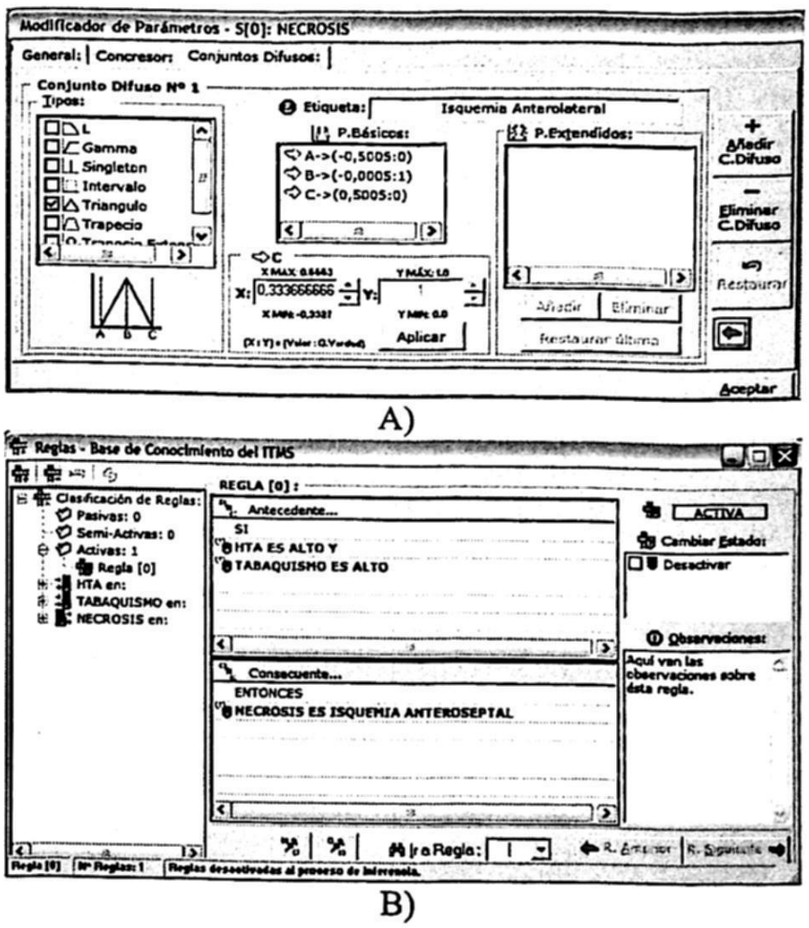


Figure 7: A) Necrosis Output Variables. Linguistic Isquemia Labels Anterolateral. B) Rule of Knowledge.

As seen in this case, the use of vague data is much used in complex systems with information from doctors in medicine, among others. The DW are of great advantage when comes the time to make decisions about certain thematic. The use of fuzzy logic can provide a good way to work with vague data. In this case, the use of an extension of a fuzzy related model FSQL to a fuzzy DW model has been shown, besides using fuzzy inference.

4 Conclusion and Future Work

The methodology based on a set of steps presented here may serve as a guide for similar toils, mainly to create data storehouses that allow to handle the information management with vague data by means of the theory of fuzzy ensembles.

Proposing the use of Type 1, 2 and 3 attributes, linguistic labels and inference rules is a contribution to supporting the decision making and, in some instances, satisfy

more efficiently the user's requirements than the traditional DW systems. FSQL created by Galindo can be incorporated to different SQL platforms with related models (Escobar, 2003), including data stores, o DW (Salas,2006).

One of the tasks is to represent in a concept model for DW that includes Type 1, 2 and 3 attributes, besides the linguistic labels shown on this work. Produce a clear method to specify the user's requirements when considering imprecise data which could be handled with fuzzy logic.

Acknowledgments: Catholic University of the Maule Internal Project Number 81201(2006-2007).

References

1. Carpani Fernando (2000), *"CMDM: Un modelo conceptual para la Especificación de Bases Multidimensionales"*, Tesis de Maestría, Universidad de La Republica, Uruguay.
2. Escobar Calixto (2003), *"Software para control difuso de todo tipo de sistemas (SCD): Aplicación al Control de Invernaderos Industriales"*. Tesis de título de Ingeniero Técnico Industrial (Electrónica).
3. Galindo J., Urrutia A., Piattini M. (2005): *"Fuzzy Databases: Modeling, Design and Implementation"*. To publish by Idea Group Publishing Hershey, USA.
4. Galindo J. (1999): *"Tratamiento de la Imprecisión en Bases de Datos Relacionales: Extensión del Modelo y Adaptación de los SGBD Actuales"*. Ph. Doctoral Thesis, University of Granada (Spain). (www.lcc.uma.es).
5. Golfarelli, M. Rizzi, S.:"Methodological Framework for Data Warehouse Design.", DOLAP'98, USA,1998.
6. Ling Feng, Tharam S. Dillon, Fellow, IEEE (2003), *"Using Fuzzy Linguistic Representations to Provide Explanatory Semantics for DataWarehouses"*, IEEE Transactions on Knowledge and Data Engineering, Vol. 15, N° 1.
7. Inmon, W (1996): *"Building the Data Warehouse"*. John Wiley & Sons, Inc. 1996.
8. Kimball, R.: *"The Datawarehouse Toolkit"*. John Wiley & Son, Inc., 1996.
9. Kumar Pavan, Krishna Radha, Kumar Supriya (2005), *"Fuzzy OLAP Cube for Qualitative"* Institute for Development and Research in Banking Technology, IDRBT-2005
10. Marotta, A.: *"Data Warehouse Design and Maintenance through Schema Transformations"*. Master Thesis. Advisor: Raúl Ruggia. Pedeciba, Universidad de la República, Uruguay, 2000.
11. Medina J.M., Pons O., Vila M.A. (1995): *"FIRST. A Fuzzy Interface for Relational SysTems"*. VI International Fuzzy Systems Association World Congress (IFSA'1995). Sao Paulo (Brasil).
12. Peralta Verónika (2001), *"Diseño Lógico de Data Warehouse a partir de Esquemas Conceptuales Multidimensionales"*, Tesis de Maestría, Universidad de La Republica, Uruguay.
13. Salas Yosselin (2006): *"Extensión del Diseño e Implementación de un Data Warehouse Difuso, Aplicado a Enfermedades Cardiovasculares"*. Tesis para Ingeniero en Computación e Informática Universidad Católica del Maule, Chile.
14. Urrutia A. (2003): *"Definición de un Modelo Conceptual para Bases de Datos Difusas"*. Ph. Doctoral Thesis, University of Castilla-La Mancha (Spain).
15. Zadeh L.A., *"Fuzzy sets"*. Information and Control 8, pp 338-353, 1965.

Optimization

A Simulated Annealing Approach for Multistage Portfolio Optimization

Maria A. Osorio, Erika C. Jimenez, A. Sánchez L. and Miguel A. Gómez*

Facultad de Ciencias de la Computación, BUAP
14 Sur y San Claudio, 72570 Puebla, Pue. México
*Universidad de las Américas Puebla, Cholula, Pue. Mexico

Abstract. This paper describes the application of a simulated annealing approach to find an optimal investment strategy by maximizing expected wealth at the end of a multistage horizon, considering wealth, return, cash balance, withdrawals and upper bounds. The discretization of the random return values and its probability was represented in a scenario tree generated with simulation and randomized clustering. The performance of the linear optimization model on different scenario trees is illustrated using test examples. The model is stochastic, and exact optimization algorithms may have difficulties with large or complex instances, motivating the research of heuristic techniques. The computational results indicate that the approach is promising for this sort of problems because easily allow the introduction of more specific and real conditions, as constraints in the model.

1 Introduction

In financial portfolio management, multistage stochastic programming is used to find an optimal investment strategy by maximizing expected wealth at the end of the planning horizon taking in account the possible fluctuation of the assets return in the future (Trippy et al. [9]). The uncertainty on return values of instruments is accurately described by a continuous distribution represented by a discrete approximation. Given history up to the commencement of the investment period, the determination of the finitely many outcomes of the random return variables is called scenario tree generation. The discretization of the random values and the occurrence probability constitute a scenario tree (see Gülpinar et al [3]).

A Linear Programming (LP) model to maximize the expected wealth at the end of the investment horizon can be easily build. Expected wealth is calculated as the total net redemption value at time period T. The model is multistage because it uses the wealth generated in the previous period in order to represent the constraint in the next period. It takes into account the uncertainty of the assets return, based in the history of each asset, representing them in a scenario tree (as in Osorio et al [8]).

In spite of its theoretical interest, the basic portfolio optimization model is often too simplistic to represent the complexity of real-world portfolio selection problems in an adequate fashion. In order to enrich the model, we need to introduce more realistic constraints that involve withdrawals, diversification constraints and left open the option to include specific conditions for different applications. This is the context

© S. Torres, I. López, H. Calvo. (Eds.)
Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 65-76

Received 23/02/07
Accepted 08/04/07
Final version 21/04/07

where good metaheuristic techniques become important. In particular, Simulated Annealing has demonstrated to be an efficient and promising technique that can candle the complex models in an adequate way (Crama and Schyns [1]).

The rest of the paper is organized as follows. The Multistage Optimization in Portfolio Management theory and models is described in section 2. Section 3 includes a complete description of the Simulated Annealing procedure used. Computational examples are presented in section 4, and conclusions in section 5.

2 Multistage Optimization in Portfolio Management

In financial portfolio management, multistage stochastic programming is used to find an optimal investment strategy by maximizing expected wealth subject to constraints specified by the investor [8]. The uncertainty on return values of instruments is represented by a discrete approximation. Given history up to the commencement of the investment period, the determination of the finitely many outcomes of the random return variables is called scenario tree generation. Generating scenario trees is important for the performance of the multistage stochastic programming. The root node of the scenario tree represents the decision "today" and the nodes further on represent conditional decisions at later stages. The arcs linking the nodes represent various realizations of the uncertain variables. The dynamics of decision making is thus captured as decisions are adjusted according to realizations of uncertainty.

We use a multistage approach to the portfolio management problem to obtain a return-efficient multistage portfolio. The main concern of this paper is to find an optimal investment strategy using different asset allocations over a given finite investment horizon. Uncertainty on asset performances (or returns) is represented with a scenario tree generated by simulation. The performance of the linear optimization model on different scenario trees is illustrated using test examples.

2.1 Uncertainty Representation and Scenario Trees

Coherent uncertainty representation is a requirement for this type of models. The uncertainty is usually expressed in terms of multivariate continuous distributions. In order to represent the continuous distributions, the decision model is generated with internal sampling or a discrete approximation of the underlying continuous distribution. The random variables are the uncertain return values of each asset on an investment. The discretization of the random values and the probability space leads to a framework in which a random variable takes finitely many values. At each time period, new scenarios branch from the old, creating a scenario tree. Scenario trees can be generated based on different probabilistic approaches as simulation or optimization as presented in Gulpinar et al. [3].

Scenario trees can have different structures as shown in Fig. 1. For this research we took the last option of every parent having two branches.

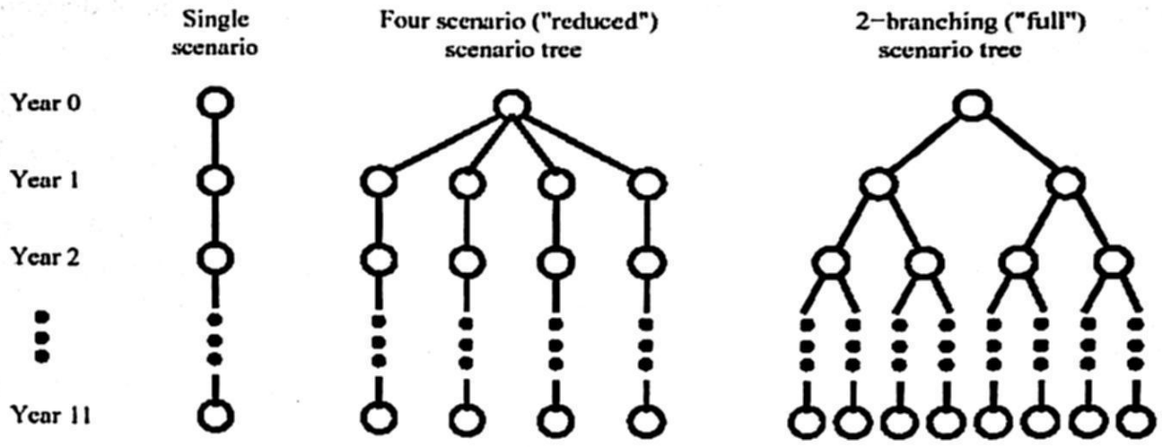


Fig. 1 Scenario Trees for Multiperiod Optimization

We assumed a portfolio of n risky assets and consider its optimal restructuring over a period in terms of expected return. After the initial investment (t=0), the portfolio may be restructured at discrete times t = 1, ..., T-1, and redeemed at the end of the period , (t=T).

Let the increasing σ-field F_t ($F_1 \subseteq \ldots \subseteq F_T$) be generated by stochastic events $\boldsymbol{\rho}^t \equiv \{\rho_1 \ldots \rho_t\}$; t = *1*, ..., *T*. Let the random variables $\mathbf{r}_t(\boldsymbol{\rho}^t)$ and $\mathbf{g}_t(\boldsymbol{\rho}^t)$ denote the uncertain dividend (or income) and capital gain returns on investment. Random variables and some specified coefficients of constraints are assumed to be F_t measurable functions ($\mathbf{r}_t$, $\mathbf{g}_t$: $\Omega_t \rightarrow \Re^n$) on some probability space (Ω_t, F_t, P_t). Due to the recourse nature of the multistage problem, decision variables $\mathbf{w}_t$, $\mathbf{b}_t$, and $\mathbf{s}_t$ are influenced by previous stochastic events $\boldsymbol{\rho}^t$, and hence $\mathbf{w}_t = \mathbf{w}_t(\boldsymbol{\rho}^t)$, $\mathbf{b}_t = \mathbf{b}_t(\boldsymbol{\rho}^t)$ and $\mathbf{s}_t = \mathbf{s}_t(\boldsymbol{\rho}^t)$. However, for simplicity, we shall use the terms $\mathbf{w}_t$, $\mathbf{b}_t$, and $\mathbf{s}_t$, and assume their implicit dependence on $\boldsymbol{\rho}^t$. We assume that ρ_t can take only finitely many values. Thus, the factors driving the risky events are approximated by a discrete set of scenarios or a sequence of events. Given the event history up to a time t, $\boldsymbol{\rho}^t$, the uncertainty in the next period is characterized by finitely many possible outcomes for the next observation ρ_{t+1}. This branching process is represented using a scenario tree.

A scenario is defined as a possible realization of the stochastic variables $\{\rho_1, \ldots, \rho_T\}$. Hence, the set of scenarios corresponds to the set of leaves of the scenario tree, N_T, and nodes of the tree at level $t \geq 1$ (the set N_t) correspond to possible realization of $\boldsymbol{\rho}^t$. We denote a node of the tree (or **event**) by $\mathbf{e} = (s,t)$, where s is a scenario (path from root to leaf), and time period t specifies a particular node on that path. The root of the tree is $\mathbf{0} = (s,0)$ (where s can be any scenario, since the root node is common to all scenarios). The ancestor (parent) of event $\mathbf{e} = (s,t)$ is denoted $a(\mathbf{e}) = (s, t-1)$, and the branching probability $p_\mathbf{e}$ is the conditional probability of event $\mathbf{e}$, given its parent event $a(\mathbf{e})$. The path to event $\mathbf{e}$ is a partial scenario with probability $P_\mathbf{e} = \prod p_\mathbf{e}$ along that path. Since probabilities $p_\mathbf{e}$ must sum to unity at each individual branching, probabilities $P_\mathbf{e}$ will sum up to unity across each layer of tree-nodes N_t for t = 0,1, ...,T.

Each node $\mathbf{e} \in N_t$ at a level t = *1*, ..., T corresponds to a decision $\{\mathbf{w}_\mathbf{e}, \mathbf{b}_\mathbf{e}, \mathbf{s}_\mathbf{e}\}$ which must be determined at time t, and depends in general on $\boldsymbol{\rho}^t$ and the past decisions $\{\mathbf{w}_j,$

$\mathbf{b}_j$, $\mathbf{s}_j\}$, for $j = 1, \ldots, t-1$. This process is adapted to ρ^t as $\mathbf{w}_t$, $\mathbf{b}_t$, $\mathbf{s}_t$ cannot depend on future events $\rho_{t+1} \ldots \rho_T$ which are not yet realized.

2.2 Scenario Trees Generated by a Simulation and Randomized Clustering

The scenario tree is the input to the financial optimization problem. The basic data structure is the scenario tree node, which contains a cluster of scenarios (vectors in Rn), one of which is designated as the centroid. The final tree consists of the centroids of each node and their branching probabilities.

We used the main steps used to generate the scenario tree according to Gülpinar et al. [3]. These steps are:

Step 1: (Initialization) Create a root node, with N scenarios. Initialize all the scenarios (including the centroid) with the desired starting point ("today's" prices). For a job queue consisting of the root node.

Step 2: (Simulation) Remove a node from the job queue. Simulate one time period of growth (from "today" to "tomorrow") in each scenario.

Step 3: (Randomized seeds) Randomly choose a number of distinct scenarios around which to cluster the rest: one per desired branch in the scenario tree.

Step 4: (Clustering) Group each scenario with the seed point to which it is the closest. If the resulting clustering is unacceptable, return to step 3.

Step 5: (Centroid selection) For each cluster, find the scenario which is the closest to its center, and designate it as centroid.

Step 6: (Queuing) Create a child scenario tree node for each cluster (with probability proportional to the number of scenarios in the cluster), and install its scenarios and centroid. If the child nodes are not leaves, append to the job queue. If the queue is nonempty, return to step 2. Otherwise, terminate the algorithm.

2.3 Definitions and Notation

Portfolio: A set of assets available for the investor.

Assets: The assets considered are Equities in the Mexican Bursaries Market (BMV), available for the constitution of a portfolio distribution.

Returns: Percentage of returns in the form of dividends for equities.

Net Redemption Value: Total amount of money received at the end of the horizon, when a the investment is encashed.

The notation used in the following definitions is described in Table 1. All quantities in boldface represent vectors in $\Re^n$. The transpose of a vector is denoted with the symbol $'$. In Table 1, subscript * indicates that vectors have two indices. The first index represents assets $i = 1,2, \ldots, n$. The second one denotes each event $\mathbf{e} \in N_t$ at time $t = 1, \ldots, T$ of the scenario tree.

Table 1. Notation

Symbols and Input Data	
$\mathbf{1}$	$\equiv (1,1,1, \ldots, 1)'$
$\mathbf{p} \circ \mathbf{q}$	$\equiv (p_1q_1, p_2q_2, \ldots, p_nq_n)'$ (Hadamard product)
$\mathbf{p}'\mathbf{q}$	$\equiv p_1q_1 + p_2q_2 + \ldots + p_nq_n$ (Inner product)
$\mathbf{e} \equiv (s,t)$	index denoting an event (a node of the scenario tree)
$a(\mathbf{e})$	ancestor of event **e** (parent in the scenario tree)
N_t	set of nodes of the scenario tree at time t
$p_\mathbf{e}$	branching probability of event **e**: $p_\mathbf{e} = \text{Prob}[\mathbf{e} \mid a(\mathbf{e})]$
$P_\mathbf{e}$	probability of event **e**: **if** $\mathbf{e} = (s,t)$, then $P_\mathbf{e} = \prod_{i=1 \ldots t} p_{(s,i)}$
n	number of investment assets
M	amount of initial investment
T	investment planning horizon
TW_t	total withdrawal at time t
ic_i	percentage paid in initial cost for asset i
ac_i	percentage paid in annual cost for asset i
$\mathbf{r}_{i\mathbf{e}}$	dividends or income returns for asset i at node **e**
tc	transaction cost
$\mathbf{w}^u_{i\mathbf{e}}$	upper bound for asset i
Decision Variables	
NR	net redemption value
$\mathbf{w}_\bullet$	amount of money held in each asset
$\mathbf{h}_\bullet$	withdrawal
$\mathbf{b}_\bullet$	amount bought of each asset
$\mathbf{s}_\bullet$	amount sold of each asset

2.4 Multistage LP Problem

The Linear Programming (LP) model maximizes the expected wealth at the end of the investment horizon. Expected wealth is calculated as the total net redemption value at time period T.

The redemption value is basically defined as the amount of money received at time T when the investment is encashed. The basic LP model only includes constraints to express the wealth return and cash balance. We added annual bank fees, transaction costs for purchase operations, the withdrawal variable in the wealth return equation, the total withdrawal (TW_t) equation in the model and the upper bounds on the assets amount in a diversification constraint in order to obtain a more complete and descriptive model. The constraints in the LP model are:

Net Redemption Value of every asset. **(1)**
Initial Allocation. **(2)**
Cash Balance Equations. **(3)**
Wealth for asset i in node **e**. **(4)**
Total Withdrawal at time t. **(5)**

Diversification constraints (6)

The model is multistage because it uses explicitly the wealth generated in the previous period in order to obtain the wealth in the next period. It takes into account the uncertainty of the assets return, based in the history of each asset and represent it in a scenario tree. The objective function is the sum of the net redemption values of every asset at the end of the complete horizon, i.e. the net amount of money that the investor can obtain when the total investment is encashed. The general expression for the multistage portfolio optimization model is:

$$\max \Sigma_{i=1,n} NR_i .$$

Subject to

$$NR_i = \Sigma_{e \in NT} Pe\ [1'\ \mathbf{w}_{ie}] \qquad i=1,\ldots,n \qquad (1)$$

$$\Sigma_{i=1,n} 1'\mathbf{w}_{i0} = M \qquad (2)$$

$$1'\mathbf{b}_{ie} - 1'\mathbf{s}_{ie} = 0 \qquad e \in N_t, t=1,\ldots,T,\ i=1,\ldots,n \qquad (3)$$

$$\mathbf{w}_{ie} = (1 - ac_i)\ [(1+\mathbf{r}_{ie})\mathbf{w}_{ia(e)}] - \mathbf{h}_{ie} + (1 - tc)\mathbf{b}_{ie} - \mathbf{s}_{ie} \qquad e \in N_t, t=1,\ldots,T, i=1, \ldots, n \qquad (4)$$

$$TW_t = \Sigma_{e \in Nt} Pe\ \Sigma_{i=1,n}\ 1'\mathbf{h}_{ie} \qquad t=1,\ldots,T \qquad (5)$$

$$\Sigma_{i=1,n}\ \mathbf{w}_{ie} \leq \mathbf{w}^u_{ie}\ \Sigma_{i=1,n}\ (1'\mathbf{w}_{ie}) \qquad e \in N_t, t=1,\ldots,T \qquad (6)$$

$$NR_i \geq 0 \qquad i=1,\ldots,n$$

$$\mathbf{w}_{ie},\ \mathbf{b}_{ie},\ \mathbf{s}_{ie} \geq 0 \qquad e \in N_t, t=1,\ldots,T, i=1, \ldots, n$$

Notice that the annual bank fees deducted by term $(1 - ac_i)$ for $i = 1, \ldots, n$ must be augmented by the bank's initial setup fees in the first year. For children of the root scenario node, $\mathbf{e} \in N_1$, the term becomes $(1 - ic_i - ac_i)$, and is imposed on all constraints. The wealth in every period t for asset i, is $\Sigma_{e \in Nt} Pe\ (1'\mathbf{w}_{ie})$, for $i=1,\ldots,n$, and $t=1,\ldots,T$. The total wealth in for every period can be evaluated as $\Sigma_{e \in Nt} Pe\ \Sigma_{i=1,n}\ 1'\mathbf{w}_{ie}$, for $t=1,\ldots,T$.

The number of variables and constraint in the LP model is increased by the number of assets and the topology of the scenario tree. The size of the scenario tree depends on the depth and branching at each time period. Our computational results show that even for large scenario trees it is possible to find solutions near to the optimal in a reasonable amount of time.

3 Simulated Annealing for Financial Investments

Simulated annealing is a generalization of a Monte Carlo method for examining the equations of state and frozen states of n-body systems. The concept is based on the manner in which liquids freeze or metals recrystalize in the process of annealing. In an annealing process a melt, initially at high temperature and disordered, is slowly cooled so that the system at any time is approximately in thermodynamic equilibrium. As cooling proceeds, the system becomes more ordered and approaches a "frozen" ground state at T=0. The original Metropolis scheme was that an initial state of a thermodynamic system was chosen at energy E and temperature T, holding T constant the initial configuration is perturbed and the change in energy dE is computed. If

the change in energy is negative the new configuration is accepted. If the change in energy is positive it is accepted with a probability given by the Boltzmann factor *exp -(dE/T)*. This processes is then repeated sufficient times to give good sampling statistics for the current temperature, and then the temperature is decremented and the entire process repeated until a frozen state is achieved at T=0.

By analogy the generalization of this Monte Carlo approach to combinatorial problems is straight forward (Kirkpatrick et al. [5]). The current state of the thermodynamic system is analogous to the current solution to the combinatorial problem, the energy equation for the thermodynamic system is analogous to at the objective function, and ground state is analogous to the global minimum. The major difficulty (art) in implementation of the algorithm is that there is no obvious analogy for the temperature T with respect to a free parameter in the combinatorial problem. Furthermore, avoidance of entrainment in local minima (quenching) is dependent on the "annealing schedule", the choice of initial temperature, how many iterations are performed at each temperature, and how much the temperature is decremented at each step as cooling proceeds.

3.1 Algorithm

The general algorithm implemented includes a population instead of only one individual solution and a final condition of reaching an expected value (see Holland [4] and Michalewicz [7]).

```
Start
Define parameters (initial_investment, initial_temperature,
        long, cooling_factor, population_size, prodi-
        gies_percentage, elitist_percentage, move-
        ment_amplitud, change_percentage, replace-
        ment_percentage, cloning_probability)
Read the Scenario Tree
Generate polulation_size initial solutions
temperature = initial_temperature
Repeat
      For each solution of the population Do
          For 0 to long Do
                Select the nodes to modify
                !Depends on change_percentage
                Modify nodes
                !Depends on movement_amplitude
                Accept or refuse the modification
      Sort the solutions according to their means
      Select prodigy solutions
      !Depends on prodigies_percentage
      Assign an amplifying factor to each to each prodigy
      Select elitists solutions !as many as population_size
      Assign an amplifying factor to each elitist
      Select the poorest solutions to replace
      !Dependes on replacement_percentage
      For each poorest solution Do
```

```
                Generate a random z value
                If z < cloning_probability Then
                    Replace the solution with a clone
                Else
                    Replace the solution by "Average Idol"
            temperature = temperature*cooling_factor
        Until best_solution.objc_value = expected_mean
        Display solution
        End
```

We knew in advance the exact solution in the examples tested, and finished the algorithm when the optimal solution was found, in order to determine the parameters combination that worked better for this kind of problems. We show these parameters in Table 6.

Parameters utilized in the main algorithm presented are described in Table 2.

Table 2. Dictionary

Parameters	Explanation
initial_temperature	It is determined by the rule that the probability of accepting a movement is near 1 for each element of the neighborhood at the beginning of the algorithm.
long	Number of times that a modified solution is generated at the same temperature, i.e., the time that the system remains in each temperature to reach a stable state.
cooling_factor	Speed to which the temperature is reduced, diminishing the probability that "bad" solutions are accepted.
population_size	Number of solutions that will be conserved in the population, this number will remain constant in each generation.
prodigies_percentage	Number of solutions, that according to their quality, will be the parents of the following generation.
elitist_percentage	Percentage of best solutions found while they are preserved.
movement_amplitud	Percentage that determines the neighborhood around some value of the solution within which this value can be moved when doing modifications.
change_percentage	Percentage of nodes of the solution that will undergo modifications.
replacement_percentage	Percentage of solutions that due to its low quality will be replaced by others of better quality, or by means of clonation or by another method available called "Average Idol".
cloning_probability	Probability of replacing a solution by another one by means of clonation.
amplifying_factor	Assigned in linearly decreasing form, of the prodigies percentage.

4 Computational Examples

The procedure was tested with two examples. In both cases, 50 monthly periods (2002-2006) were used to build a scenario tree with four future stages. The scenario tree has two branches in each node. We considered five assets in the first example and ten assets in the second one. The data correspond to real assets in the BMV (Mexican bursaries market) and were obtained from Econom@tica (financial database). The examples were tested in a Pentium IV with 1.7 GHz and 256 Mb.

The initial amount M was of 100 money units for both examples and we considered a withdrawal of TW_t=0, for t=1,...,T. The scenario trees used for the example with 5 assets and the example with 10 assets are showed in Tables 3 and 4.

Table 3. Scenario Tree for 5 Assets

Id node	Asset1	Asset2	Asset3	Asset4	Asset5	Probability	Id Father node	Stage
0	0.747	0.684	0.769	0.673	0.696	1.000	-1	0
1	0.856	0.703	1.104	0.691	0.741	0.346	0	1
2	0.655	0.667	0.485	0.657	0.658	0.654	0	1
3	0.897	0.710	1.229	0.698	0.757	0.290	1	2
4	0.774	0.689	0.853	0.677	0.707	0.710	1	2
5	0.914	0.713	1.284	0.701	0.764	0.595	2	2
6	0.964	0.722	1.438	0.709	0.785	0.405	2	2
7	0.687	0.673	0.584	0.663	0.672	0.237	3	3
8	0.797	0.791	0.692	0.904	0.680	0.763	3	3
9	0.736	0.682	0.736	0.671	0.692	0.559	4	3
10	0.504	0.641	0.020	0.632	0.597	0.441	4	3
11	0.797	0.693	0.924	0.681	0.717	0.805	5	3
12	0.695	0.675	0.610	0.664	0.675	0.195	5	3
13	0.716	0.678	0.673	0.668	0.683	0.499	6	3
14	0.757	0.685	0.798	0.674	0.700	0.501	6	3

Table 4. Scenario Tree for 10 Assets

id_node	Benchmarks_182D	Benchmarks_28D	Benchmarks_364D	Benchmarks_7D	Benchmarks_91D	America_Movil_A	America_Movil_L	Ara_Con_A31sorcio	Arca_Embotelladora	Asureste_B	probability	id_father_node	Stage
0	0.67	0.60	0.69	0.59	0.61	3.34	3.39	1.94	1.00	2.62	1.00	-1	0
1	1.08	0.14	0.98	1.12	0.96	3.18	3.41	0.93	0.50	1.82	0.23	0	1
2	1.14	1.17	0.91	0.72	0.46	1.53	4.21	1.07	1.53	1.56	0.77	0	1
3	1.24	0.02	0.37	1.79	1.35	1.87	4.24	0.43	0.83	3.61	0.44	1	2
4	0.05	0.21	0.40	2.18	0.43	4.97	0.60	0.72	0.71	2.69	0.56	1	2
5	0.29	2.16	0.78	1.44	0.31	0.82	4.54	0.80	0.17	2.11	0.77	2	2
6	1.17	0.83	0.28	1.36	0.89	0.73	5.24	0.55	2.85	2.78	0.23	2	2
7	1.87	0.00	0.48	3.46	1.17	1.99	6.57	0.33	0.81	0.00	0.96	3	3
8	0.09	0.01	0.70	3.58	2.56	0.13	2.19	0.82	0.93	5.34	0.04	3	3
9	0.01	0.27	0.29	3.92	0.76	7.86	1.11	0.23	1.40	4.94	0.42	4	3
10	0.06	0.38	0.31	3.11	0.21	6.88	0.52	0.25	0.75	5.33	0.58	4	3
11	0.14	3.28	0.51	2.37	0.07	0.89	3.44	1.26	0.17	0.71	0.01	5	3
12	0.19	2.39	0.29	1.39	0.61	0.83	7.25	1.05	0.26	4.08	0.99	5	3
13	0.09	0.77	0.47	0.18	1.33	1.43	8.54	0.49	3.06	1.12	0.35	6	3
14	1.67	1.66	0.47	2.65	1.19	0.90	3.83	0.25	3.57	1.47	0.65	6	3

The amount of money obtained and reinvested (because annual withdrawals are 0 for every period, for both examples) in every asset, can be seen in tables 5 and 6.

Table 5. Results for example with 5 assets

Asset	Stage 0	Stage 1	Stage 2	Stage 3	Stage 4
Asset 1	0	0	0	189.22	328.54
Asset 2	0	176.90	0	149.42	248.58
Asset 3	100.00	0	214.81	260.01	484.31
Asset 4	0	0	0	73.36	135.50

Table 6. Results for example with 10 assets

Asset	Stage 0	Stage 1	Stage 2	Stage 3	Stage 4
America_Movil_A	0	0	450.79	1,498.64	12,428.14
America_Movil_L	100.00	439.44	1,755.93	10,594.33	81,967.64

The same models were solved using CPLEX V 9.0, in order to adjust the simulated annealing procedure parameters for obtaining the optimal solutions. The number of iterations was the iterations needed to reach the optimal value. Figures 2 and 3 show the optimal value convergence (the net redemption value encashed at the end of the horizon) with the simulated annealing procedure presented in this paper.

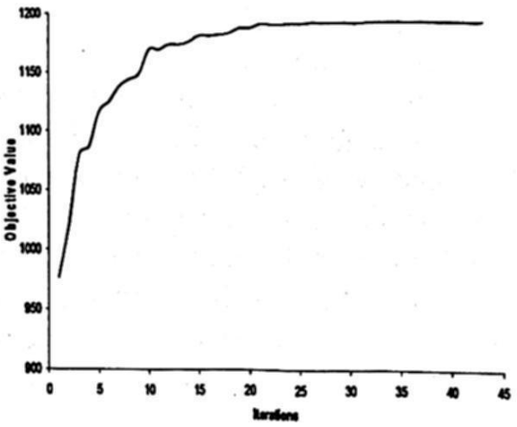


Fig. 2 Optimal value convergence for example with 5 assets

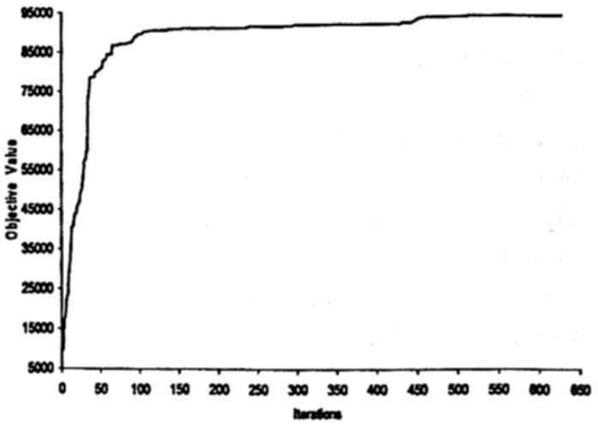


Fig. 3 Optimal value convergence for example with 10 assets

Because SA is a metaheuristic, there are many parameters to fix in order to turn it into an efficient algorithm. We have tested several parameters values to find more appropriate choices for this type of problems and the best parameters are presented in Table 7.

Table 7. Best parameter values

Parameters	5 Assets	10 Assets
initial temperature	450	1000
Cooling_factor	0.1	0.1
population_size	5	5
Long	5	5
prodigies_percentage	40%	40%
elitist_percentage	45%	45%
movement_amplitud	30	30
change_percentage	10%	10%
replacement_percentage	40%	20%
clonation_percentage	60%	40%

5 Conclusions

Portfolio selection gives rise to difficult optimization problems when realistic side constraints and variables are added to the basic model. Exact optimization algorithms cannot always deal efficiently with such complex models. It seems reasonable, therefore, to investigate the performance of heuristic approaches in this framework (Maringer et al. [6]).

Simulated annealing is a powerful tool for the solution of many optimization problems. Its main advantages over other local search methods are its flexibility and its ability to approach global optimality. The main objective of this paper was therefore to investigate the adequacy of simulated annealing for the solution of more realistic portfolio optimization models. The resulting algorithm allowed us to get the optimal net redemption value for the examples tested. The algorithm is able to handle more classes of constraints than many other approaches found in the literature.

Although there is a clear trade-off between the quality of the solutions and the time required to compute them, the algorithm can be said to be quite versatile since it does not rely on any restrictive properties of the model (Green et al. [2]). For instance, the algorithm does not assume any underlying factor model for the generation of the covariance matrix. Also, the objective function could conceivably be replaced. Nevertheless, the tailoring work required to fine-tune the parameters of the algorithm was rather delicate. Besides, introducing additional classes of constraints of new features in the model would certainly prove quite difficult again.

References

1. Crama, Y., Schyns, M.: Simulated annealing for complex portfolio selection problems. European Journal of Operational Research, Vol. 150, No. 3, November 2003, (2003) 546-571.
2. Green, R., Burton, H. (1992). When Will Mean-Variance Efficient Portfolios be Well Diversified? Journal of Finance, Vol. 5 (47), (1992) 1785-1809.
3. Gülpinar, N., Rustem, B., Settergren, R.: Optimization and Simulation Approaches to Scenario Tree Generation. Journal of Economics Dynamics and Control, Vol. 28, Issue 7 (2004) 1291-1315.
4. Holland, J. H. (1975). *Adaptation in natural and artificial systems.* Ann Arbor: The University of Michigan Press.
5. Kirkpatrick, S., Gelatt, C. D., Vechhi, P.M.: Optimization by simulated annealing. Science, Vol. 220, (1983) 671-680.
6. Maringer, D., Kellerer, H. Optimization of Cardinality Constrained Portfolios With an Hybrid Local Search Algorithm. Or Spectrum, Vol. 25(4), (2003) 481-495.
7. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. 3rd edn. Springer-Verlag, Berlin Heidelberg New York (1996).
8. Osorio, M.A., Gulpinar, Settergren, R., Rustem, B.: Post-Tax Optimization w8. Osorio, M.A., Gulpinar, Settergren, R., Rustem, B.: Post-Tax Optimization with Stochastic Programming. European Journal of Operational Research, Vol. 157 (2004) 152-168.
9. Trippy, R., Lee, J.: Artificial Intelligence in Finance and Investing: state-of-the-art technologies for securities selection and portfolio management. IRWIN Professional Publishing, USA (1996).

Image Processing

Star Fields: Improvements in Shape-Based Image Retrieval

Alberto Chávez-Aragón[1], Oleg Starostenko[2] and Leticia Flores Pulido[1]

[1] Universidad Autónoma de Tlaxcala, Calzada Apizaquito s/n km. 1.5, Apizaco Tlaxcala, México
[2] Universidad de las Américas-Puebla, Sta. Catarina Mártir Cholula, Puebla, México

Abstract. Determining the similarity of two images is a very difficult task in both machine and human vision systems. Not mention the semantic gap. Thus, in order to reduce this problem this paper developed a set of methods for retrieving images based on one low level image feature such as shape. We focused on this important feature of the objects because there is evidence that natural objects are primarily recognized by their shapes. In this paper, we proposed an alternative representation of shapes, that we have called two segment turning function (2STF) which has a set of invariant features such as invariant to rotation, scaling and translation. Then, based on 2STF, we proposed a complete new strategy for computing a similarity among shapes. This new technique was called Star Field (SF). The proposed technique, which is made up of a set of new methods, was implemented in a test-bed CBIR system that we called IRONS. IRONS stands for "Image Retrieval based ON Shape".

1 Introduction

Today huge amounts of new digital documents are available around the world. Every day different types of digital documents such as text, image, video, audio, and animation, among others, are added to the Internet or similar technologies. However, most current search engines' algorithms use text as a principal document descriptor. Techniques which use different descriptors like shape, color, sound, etc. lag behind text-based techniques. This is why there is a growing need for efficient visual information retrieval algorithms which go beyond the text-based retrieval approach. In other words, there is a lack of reliable and efficient systems to get relevant information contained in multimedia documents. This paper addresses the problem of retrieving documents that contain visual information. Although it is true that content-based image retrieval systems already exist, many of these systems some times retrieve irrelevant documents or documents unrelated to the user's query. This problem is caused by the use of low-level image descriptors; furthermore, these descriptors hardly have a semantic weight. Specifically, this work addresses the image retrieval problem based on shape, since shape has a meaning by itself.

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 79-90

Received 23/02/07
Accepted 08/04/07
Final version 23/04/07

2 Visual Information Retrieval

The visual information retrieval problem is an extension of the IR problem to the images domain. Visual Information Retrieval can be define as: *"The use of technology to obtain result images from a query based on its visual information"* [13]. In other words, its purpose is to retrieve from a database images or image sequences which are relevant to a query. The Visual Information Retrieval area (VIR) challenge is to go beyond the text search, which describes the images in order to store and recover visual information from digital repositories.

There are several reports about recent researches in the visual information retrieval area. Consequently, many VIR systems have been produced. According to Venters [2] all of these systems can be classified into two main groups: Commercial Image Retrieval Systems and Prototype Research Systems. Among commercial image retrieval applications, the following systems stand out: Excalibur Visual RetrievalWare [6], ImageFinder, IMatch [26], QBIC [22] and Virage [10]. On the other hand, AMORE [3], Photobook [16], PictoSeek [25], SQUID [23], VisualSEEK [11], Black Box [13] and Keyblock [19] are examples of Prototype Research Systems.

3 Shape-Based Retrieval

Perhaps the most obvious requirement of users for VIR systems is to retrieve images by shape, since there is evidence that natural objects are primarily recognized by their shape [9]. Features vectors which represent object shapes contained in images are computed in order to be indexed in a database. The query process works in the same way that color-based and texture-based retrieval work in the sense that a query can be an image. But, unlike color and texture retrieval, shape-based retrieval has another particular way to feed the query into the system. This is by means of sketching. Systems which support this kind of queries must provide the user with a sketch tool [12],[27].

3.1 Shape Representation and Matching

There is no universal definition of what shape is, but it is possible to mention some well accepted definitions. Shape is the outward form of an object defined by its outline; shape is the external appearance of something [7]. In this paper we consider shape as a geometrical pattern, consisting of a set of points, curves, surfaces, solids, etc. Shape matching is considered one of the most difficult aspects of content-based image retrieval since the representation of shapes is often more complex than color and texture. The difficulty lies in the fact that a common shape needs a lot of parameters to be represented explicitly.

4 Shape Representation

Traditionally, a shape is described as a closed polygon. However, the polygonal representation of shapes is not a convenient way for computing the similarity among them. In order to overcome this problem we propose a different representation that we have called two-segment turning function. Our technique is based on tangent space representation but it has some advantages that are outlined below.

4.1 Polygonal Representation

Our strategy for computing similarity among shapes starts out getting the outline of the shape from an image. Basically, we assume as a premise that each image we are working with represents just one object. Besides, the object has been previously separated from the background. That means that our images are binary ones and the objects are represented by white pixels and the backgrounds by black pixels.

The method for getting the outline consists of two main stages. The first step designates one pixel of the object border as the starting point. We choose as starting point the first white pixel which is on the first row that belongs to the object. The second step consists in tracking down those pixels that make up the object border. The tracking task is make in the clockwise direction. Figure 1 shows a result of outline detector algorithm.

Fig. 1. The image on the left is a binary image and it was the input of the outline detector algorithm The image on the right shows the result given by the algorithm.

Up to now, a closed polygon which represents the object we are interested in is obtained; however, this polygon has plenty of vertices. Thè next natural step is to reduce the number of vertices so that we can apply an efficient similarity strategy.

4.2 Relevance easure

In order to decrease the number of vertices of a shape it is necessary to calculate what is the relevance of each vertex. The relevance measure K that we use is

based on two parameters, the length and the turn angle of two consecutive line segments which share the vertex we want to compute its relevance. The relevance is defined as it is shown in equation 1.

$$K(S_1, S_2) = \frac{\beta(S_1, S_2) l(S_1) l(S_2)}{l(S_1) + l(S_2)} \tag{1}$$

where $\beta(S_1, S_2) l(S_1)$ is the turn angle at the common vertex of the segments S_1, S_2, and l is the length function normalized with respect to the total length of the polygonal curve C. The lower value of $K(S_1, S_2)$ is, the less contribution to the shape of the curve of arc $S_1 \cup S_2$ is. To stop the evolution process it is necessary to use a parameter that defines the number of iterations or to use a threshold which represents the permitted range of values for any simplified shape vertex. A curve evolution algorithm makes the former task. Figure 2 shows the results obtained after applying the curve evolution algorithm to a polygon. It is clear that curve evolution algorithm keeps the main visual parts of the original polygonal curve and obviously the amount of information has decreased drastically.

Fig. 2. The closed polygon on the left is the evolution of the original polygon, it has 30 segments. On the other hand, the figure on the right shows a stage of evolution with just 20 segments; in spite of this the main visual parts are maintained.

5 Two-segment Turning Function

The polygonal representation of a shape is not a convenient form to calculate how similar is that shape to another. In order to overcome this problem and make easier and more effective the matching process, we propose a alternative representation that we have called two-segment turning function or $2STF$. Using $2STF$ a polygonal curve P is represented by the graph of a step function, the steps on $x-axis$ represents the normalized arc length of each segment in P, and the $y-axis$ represents the turn angle between two consecutive segments in P. The former feature gives the name to our proposed technique. Figure 3 shows the angle that is taking into account in order to build the $2STF$. The angle is defined by S_2 and the imaginary line that pass through the segment S_1. This form for measuring the angle has an intuitive reason and this is that the angle measures the deviation of the second segment in respect to the first segment direction. It is clear that the angle values are in the interval $[-\pi, \pi]$.

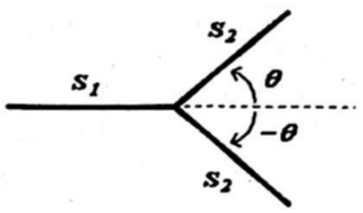


Fig. 3. This figure shows the angle used for $2STF$. The angle is defined by the imaginary line passing through the first segment and the second one. A left turn makes the angle positive and a turn in the clockwise direction makes the angle negative.

6 Star Field

Star Field (SF) is an alternative representation for shapes that allows us to apply a different algorithm in order to obtain a similarity value of two curves. This new algorithm we will propose below does not provide a way to determine the best correspondence among two functions but a very good solution. As a result, Star Field along with a new similarity algorithm are expected to give an easier and faster matching process. A Star Field formally is a torus T_1 x T_2, where T_1 is a circle of length one that represents the length of a polygonal curve and T_2 is a circle that represents the turning direction of digital steps from $2STF$. Nevertheless, most of the time we consider a SF as a window that shows a $2D$ projection of a previously processed torus. This window is made up of stars or points, that is where the name comes from, and each of them represents the relevance measure of each $2STF$ step.

6.1 From $2STF$ to SF

One of the mayor difference between the use of $2STF$'s similarity measure an the one using SF is regarding to the grade of evolution of the digital curves they work with. A star field diagram is basically a $2D$ plane, it is divided horizontally into two section. The upper section holds the stars that represents vertices of concave arcs. On the other hand, lower part holds vertices of convex arcs. Each star on SF is defined by means of two coordinates. The $y-coordinate$ represents the angle between two consecutive segments. Due to the use of $2STF$ for representing a shape, the interval of the turning angle is $[-\pi, \pi]$ radians. However, in the Star Field the angle is normalized in the interval $[0, 1]$. With respect to the $x-coordinate$, these values correspond to the accumulative length of the steps in $2STF$ from the starting point to the current point. In other words, the $x-coordinate$ represent how far is each vertex from the starting vertex and also this distance is normalized.

To illustrate the way a Star Field looks like, imagine that the $2STF$ has just decreasing steps, the Star Field representation of this function will be crowded in

the lower part. This kind of Star Fields represents mainly convex shapes. In the same way, if the $2STF$ shows raising steps, that means that it represents a mainly concave figure and the Star Field is crowded in the upper part. Finally, if a step has an angle equal to zero with respect to the previous one, the $y - coordinate$ of the corresponding star has the value .5 in the Star Field, this is because the values of the Star Field go from [0,1] in both directions. Likewise, if two consecutive segments have $-\pi$ or π radians the $y - coordinate$ of the corresponding point in the Star Field has 0 or 1 respectively. To illustrate this, consider figure 4

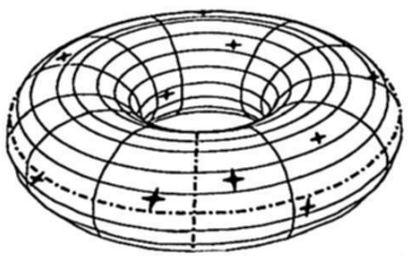


Fig. 4. Star Field, real representation. Actually, Star Fields can be seen as a bending surface like it is shown in this figure. Each star or point, in the Star Field represents the vertex that is shared by two consecutive steps from the equivalent $2STF$.

As we have mentioned before, a Star Field diagram is basically a $2D$ plane. In order to transform a torus into a 2D plane, we imaginatively cut the torus on two places following dotted-lines as it is shown in figure 5. Then, it is necessary to bend the surface, in the sense the arrows show, to get our desired 2D plane. As a result, we obtained a plane similar to the one shown in figure 6

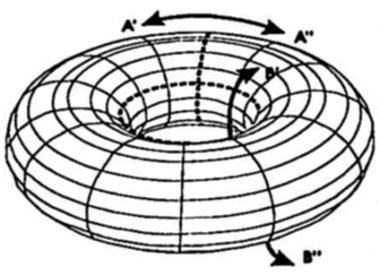


Fig. 5. This figure shows where to cut the torus and in what direction we have to bend it, so that a 2D Star Field representation is obtained.

Since Star Field is based on $2STF$, it has the same invariant characteristics as $2STF$, demonstration of those features are beyond the scope of this paper, for further details see [4].

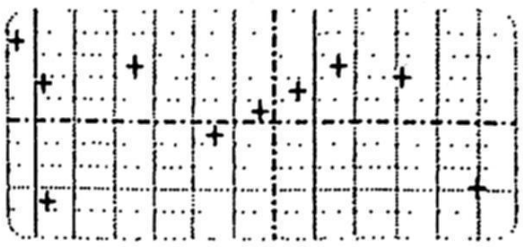


Fig. 6. 2D Star Field representation obtained by means of cutting and bending the torus in the way we explained above.

7 Matching Graph

So far, a new convenient way for representing a polygonal curve has been presented. This new representation give us an idea of who similar two polygonal curves are. However, we need a precise measure. So, we proposed a new similarity measure that makes use of a graph that has particular features. In this section the construction process of this graph is presented.

Given two polygonal curves P_1 and P_2 and their Star Field representations SF_1 and SF_2, the graph G that allow us to compute their similarity is defined as follows. $G = (V, E)$ where V and E are disjoint finite sets. We call V the vertex set and E the edge set of G. Our particular graph G has a set V which consists of two smaller subset of vertices v_1 and v_2. $V = v_1 \cup v_2$, where v_1 is the set of point of SF_1 and v_2 is the set of points of SF_2. On the other hand, E is the set of pairs (r, s), where $r \in v_1$ and $s \in v_2$.

According to previous definition the edges of our graph, that we will call from now on matching graph or MG, consists of two points and each point comes from a different Star Field representation. But also a new restriction will must be introduced, this is stated as follows. $\forall (r, s) \in E$, there is not more that one pair (r, s) that has the same point s. This restriction has an intuitive idea and this is, one point of the first curve can be matched with n points of the second one but not in the inverse sense. We have to say that the number of points of each Star Field can be different and that is because we can match polygons with different grade of evolution.

7.1 Matching raph

The main idea behind the construction of the matching graph consist in building a connected weighted graph so that an algorithm to find the minimal spanning tree is applied. The minimum spanning tree is a subset of edges that forms a tree that includes every vertex, where the total weight of all the edges in the tree is minimized. This way, the lower value of total weight the more similar are the shapes involved. But, in order to get the desired result the matching graph

must be constructed in a very particular way. This method of construction is shown in the Matching graph construction algorithm .

Matching graph construction

input: two set of points SF_1 and SF_2 that define the two Star Field representations, an increment Δ and a distance d
output: a connected weighted graph

1. rotate in the x direction SF_1 and SF_2 so that, the most import star of each SF coincides in the center of the window
2. for each point sf_1pn from the SF_1 do
3. look for those points that belong to SF_2, that stay at most a distance d in all directions from sf_1pn and that have not been connected previously
4. connect sf_1pn with each point found in previous step and assign a weigh equal to the euclidian distance of the two vertices to each edge
5. if there wasn't any connection, increase d in a value Δ and go to step 3
6. Select one point of SF_1 and connect the rest of the points from SF_1 with it; finally assign each edge generated in this step a weigh equal to zero

Given two identical shapes with the same number of steps, the total weight of the spanning tree is equal to zero. This is, because each star is connected with the corresponding one and since they have the same value of $x - coordinate$ and $y - coordinate$ the euclidian distance is equal to zero. Additionally, we have mentioned that all the stars from the first shape are connected with a weight equal to zero. As a result, the values of the path through the spanning tree is zero, that means that they are identical. The algorithm for finding the minimum spanning tree most of time is called Prim's algorithm.

7.2 Similarity easure

Finally, we can define how to calculate the similarity among shapes. The most important part of this calculation is the value of the cumulative weight of the edges that make up the spanning tree. However, the similarity value is also affected by a penalty quantity, this is because some stars have not been connected with the corresponding ones.

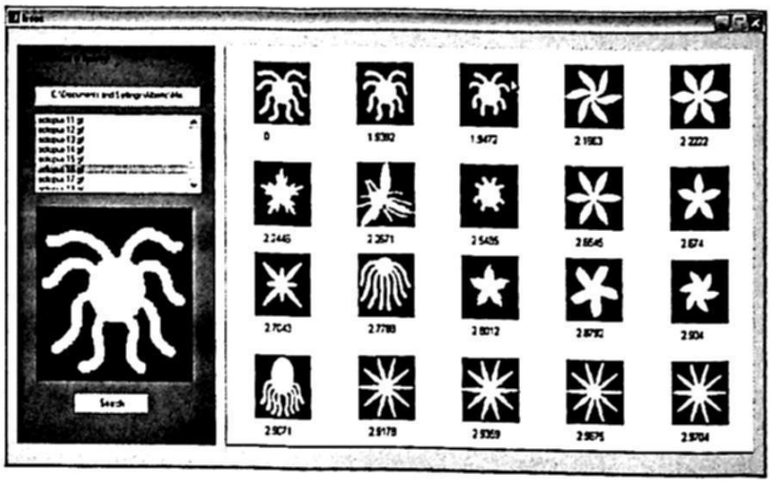


Fig. 7. IRONS' GUI, IRONS was developed using Matlab

8 Results

In the majority of the experiments of this paper we used the database CE-Shape-1 [21]. The reason why we selected this image database is because this set of images has been used for testing similar works, this allows us to have a reference framework to compare with. The Core Experiment CE-Shape-1 for shape descriptors performed for the MPEG-7 standard consists of 1400 images divided into 70 classes with 20 images each. A single image is a simple pre-segmented shape defined by their outer closed contour. Since the $2D$ objects are projections of $3D$ objects their silhouettes may change due to:

- change of a view point with respect to objects
- non-rigid object motion (e.g.people walking or fish swimming)

Table 1 describes shortly a set of shape descriptors which were tested in Core Experiment CE-Shape-1 and these works are the ones we compare with our proposed method.

First experiment consist in verifying how robust is our method with respect to scaling and rotation changes. We done this experiment in the way is described in part A of MPEG-7 standard experiments. Results are shown in table 2. Our method is labeled as **G**.

We can say that our method is robust to changes in scaling and rotation as we have already demonstrated comparing our method with those of the MPEG-7 core experiment. We cannot forget that the 91.40% was obtained in a very strict experiment and this value is not far from those reported by the the MPEG-7 core experiment and in some cases even better.

Table 1. Shape descriptors which were tested in the Core Experiment CE-Shape-1

Descriptor	Type of descriptor	Presented by	Technology
A	contour based descriptor	Mitsubishi Electric ITE-VIL	based on the curvature scale-space [19], [20]
B	contour based descriptor	Henry Hertz Institute in Berlin	based on wavelet representation of object contours [21]
C	contour based descriptor	Longin Jan Latecki and Rolf Lakämper in cooperation with Siemens Munich	best possible correspondence of visual parts [22], [23]
D	image based descriptor	Hanyang University	based on Zernike moments [24]
E	image based descriptor	Hyundai Electronics Industries	based on multilayer eigenvectors [18]
F	skeleton based descriptor	Mitsubishi Electronic and Princeton University	tree-matching algorithm [25], [26], [27]
G	contour based descriptor	The authors of this paper	Star fields

8.1 Similarity Based Retrieval

The retrieval rate consists in computing the number of correct matches in the top 40 retrieved images using a single image from any class as image query. There are some images in a single class that are semantically related but numerically unrelated, this is why, it is not possible to have a 100% retrieval rate. The retrieval rate of the descriptor is near to 72% table 3 shows the precise figures.

Additional experiments using our proposed technique to retrieve images as well as the IRONS system are described in detail in [4].

9 Conclusions

We proposed a complete new strategy for computing a similarity among shapes. This new technique was called Star Field (SF). Star Field inherits from $2STF$ invariant characteristics. Additionally, Star Field allows us to work with less simplified digital polygons; since, it permits to define a similarity measure based on the calculation of a minimum spanning tree from a connected weighted graph. Among the outstanding points of our set methods we can mention: ease of use and implement, it uses visual parts as a parameter of similarity like humans do, it has a good performance as we demonstrated in this paper.

The proposed technique, which is made up of a set of new methods, was implemented in a test-bed CBIR system that we called **IRONS**. IRONS stands for "**I**mage **R**etrieval based **ON** **S**hape". IRONS was developed using the Matlab

Table 2. Robustness to scaling and rotation, our results are labeled as **G**

Shape descriptor	Invariant to scaling	Invariant to rotation	Robustness to scaling and rotation
A	89.76	99.37	94.56
B	88.04	97.46	92.75
C	88.65	100.00	94.32
D	92.54	99.60	96.07
E	92.42	100.00	96.21
F	no results	no results	85
G	**91.78**	**93.05**	**91.40**

Table 3. Part B results of the Core Experiment CE-Shape-1, our proposed method is labeled as G

Shape descriptor	Similarity-based retrieval Percentage of correct matches
A	77.44
B	67.76
C	76.45
D	70.22
E	70.33
F	60
G	**71.82**

language. IRONS is just a prototype for testing our proposal image retrieval technique, and it does not pretend to be a full operational system.

To conclude, we proposed an high effective, ease to implement and robust image retrieval technique which uses the shapes of the objects as a main descriptor. Our approach is comparable in results with those systems which compute the best correspondence among shapes. However, our approach does not attend to find the best correspondence but it finds a very good approximation.

References

1. H. Blum. Biological shape and visual science. *Journal of Theor. Biol.*, 38:205–287, 1973.
2. V. C. C. and C. M. D. A review of content-based image retrieval systems. JISC Technology Applications Programme, 2000.
3. N. U. I. CC Research Laboratories. The amore, advance multimedia oriented retrieval engine. http://www.ccrl.com/amore/, 2002.

4. J. A. Chávez-Aragón. *Star Field Approach for Shape-Based Image Retrieval: Development, Analysis and Applications.* Ph.d. Thesis, Universidad de las Américas-Puebla, Mexico, 2007.
5. G. Chuang and C.-C. Kuo. Wavelet descriptor of planar curves: Theory and applications. *IEEE Trans. on Image Processing*, 5:56–70, 1996.
6. E. corp. Excalibur visual retrievalware software deveopers kit. 2004.
7. M. S. L. (Ed.). *Principles of Visual Information Retrieval.* Springer-Verlag London, 2001.
8. S. A. F. Mokhtarian and J. Kittler. Efficient and robust retrieval by shape content through curvature scale space. *A. W. M. Smeulders and R. Jain (ed), Image Database and Multimedia Search, World Scientific Publising, Singapore*, pages 51–58, 1997.
9. B. I. Recognition-by-components: a theory of human image undestanding. *Psychological Review*, 94(2):115–147, 1987.
10. V. Inc. Vir image & image read/write toolkit. http://www.virage.com/, 2004.
11. S. J. and C. S. Visual seek: a fully automated content-based image query system. *proceedings of ACM Multimedia 96*, 1:87–98, 1996.
12. H. K. and K. T. Query by visual example - content-based image retrieval. *EDBT'92 Third International Conference on Extending Database Technology*, 1:56–71, 1992.
13. S. K., S. A., and M. S. Black box approach to image feature manipulation used by visual information retrieval engines. University of Georgia, Department of Computer Science, technical report, 1997.
14. S. J. D. K. Siddiqi, A. Shokoufandeh and S. W. Zucker. Shock graphs and shape matching. *Int. J. of Computer Vision*, 2000.
15. A. Khotanzan and Y. H. Hong. Invariant image recognition by zernike moments. *IEEE Trans. PAMI*, 12:489–497, 1990.
16. M. M. Lab. Photobook. http://www-white.media.mit.edu/vismod/, 2003.
17. L. J. Latecki and R. Lakämper. Contour-based shape similarity. *Proc. of Int. Conf. on Visual Information Systems*, LNCS 1614:617–624, 1999.
18. L. J. Latecki and R. Lakämper. Shape similarity measure based on correspondence of visual parts. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2000.
19. Z. Lei, R. Aibing, and Z. Aidong. Advanced feature extraction for keyblock-based image retrieval. *Proceedings of the 2000 ACM workshops on Multimedia*, 1, 2000.
20. I.-J. Lin and S. Y. Kung. Coding and comparison of dags as a novel neural structure with application to on-line hadwritten recognition. *IEEE Trans. Signal Processing*, 1996.
21. R. L. Longin J. Latecki and Eckhardt. Shape descriptors for non-ridig shape with a single closed contour. *CVPR 2000*, 2000.
22. F. M. and S. H. Query by image and video content: The qbic system. *IEEE computer*, 28(9), 1995.
23. F. Mokhtarian. Shape queries using image databases. http://www.ee.surrey.ac.uk/Research/ VSSP/imagedb/demo.html, 2002.
24. F. Mokhtarian and A. K. Mackworth. A theory of multiscale, curvature-based shape representation for planar curves. *IEEE Trans. PAMI*, 14:789–805, 1992.
25. G. Th. and S. A. W. M. The pictoseek www image search system. *IEEE ICMCS*, 1, 1999.
26. M. M. Westphal. Imatch, mwlabs. http://www.mwlabs.de/, 2004.
27. C. Y. and K. S. Y. A hierarchical algorithm for image retrieval by sketch. *First IEEE Workshop on Multimedia Signal Processing*, 1:564–569, 1997.

3D Rank Median L-Filters to Process Video Sequences

Jose Luis Varela-Benitez[1], Francisco Gallegos-Funes[1],
Rene Cruz-Santiago[1] and Volodymyr Ponomaryov[2]

[1] Av. IPN s/n, U.P.A.L.M. SEPI-ESIME, Edif. Z, Acceso 3, Tercer Piso, SEPI Electronica,
Col. Lindavista, 07738, Mexico, D. F., Mexico
fgallegosf@ipn.mx

[2] ESIME-Culhuacan; Av. Santa Ana 1000, Col. San Francisco Culhuacan,
04430, Mexico D.F., Mexico
vponomar@ipn.mx

Abstract. This paper presents the Rank Median L-Filters to suppress speckle noise in the 3D ultrasound sequences. The proposed technique uses the Rank M-type (RM) estimator and this one is adapted to 3D video processing applications. The real-time implementation of the proposed algorithm is realized by means of use of the DSP TMS320C6711. Therefore, the results from known 3D techniques are compared with the proposed one to demonstrate its performance in terms of noise suppression, detail preservation, and processing time.

1 Introduction

The 3D ultrasound imaging has been considered as one of the most powerful techniques for medical diagnosis and it is often prefer over other medical imaging modalities due it is noninvasive, portable, and versatile [1-3]. It does not use ionizing radiations, and is relatively low-cost. One of the areas where research in this field has addressed is the fundamental problem of speckle noise influence, which is a major limitation on image quality in ultrasound imaging [1, 2].

Imaging speckle is a phenomenon that occurs when a coherent source and a noncoherent detector are used to interrogate a medium, which is rough on the scale of the wavelength. Speckle noise occurs especially in images of the liver and kidney whose underlying structures are too small to be resolved using long ultrasound wavelength. The presence of speckle noise affects the human interpretation of the images as well the accuracy of computer-assisted diagnostic techniques. As a result, speckle filtering is a critical pre-processing step for feature extraction, analysis, and recognition from medical imagery measurements [1, 2].

In this paper, we present the capability and real-time processing features of the robust RM-L (Rank M-type L) filters [4] for the removal of speckle noise in 3D ultrasound images. An experimental system was used to capture 3D ultrasound images. The Texas Instruments DSP TMS320C6711 was used to implement the algorithm and to obtain the processing time [5, 6]. Different configurations of sweeping cubes (voxels) were used to obtain a balance between the processing time and quality of the res-

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 91-100

Received 09/02/07
Accepted 15/04/07
Final version 20/04/07

toration of the 3D images [7, 8]. Extensive simulation results have demonstrated that the proposed filter can consistently outperforms other filters by balancing the tradeoff between noise suppression, detail preservation, and processing time.

2 The Speckle Noise

A general model for ultrasound speckle noise can be written as [2],

$$x(i,j)=S(i,j)\eta_m(i,j)+\eta_a(i,j) \tag{1}$$

where $x(i,j)$ is a noisy observation (i.e., the recorded ultrasound image) of the two-dimensional (2D) function $S(i,j)$ (i.e., the noise-free image that has to be recovered), $\eta_m(i,j)$ and $\eta_a(i,j)$ are the corrupting multiplicative and additive speckle noise components, respectively, and i and j are variables of spatial locations that belong to 2D space of all real numbers $(i,j)\in\Re^2$.

Generally, the effect of the additive component (such as sensor noise) of the speckle in ultrasound images is less significant than the effect of the multiplicative component (coherent interference). Thus, ignoring the term $\eta_a(i,j)$, one can rewrite (1) as [2]

$$x(i,j)=S(i,j)\eta_m(i,j) \tag{2}$$

To transform the multiplicative noise model into additive one, we apply the logarithm function on both sides of (2) [2]

$$\begin{aligned}\log x(i,j)&=\log S(i,j)+\log\eta_m(i,j)\\ x^l(i,j)&=S^l(i,j)\eta_m^l(i,j)\end{aligned} \tag{3}$$

where $\eta_m^l(x,y)$ is approximated as additive white noise. We assume here that the speckle pattern has a white Gaussian noise model.

3 3D Rank M-type L-Filters

In recent works [9, 10], we proposed the combined RM (Rank M-type) –estimators for applications in image noise suppression. These estimators use the *M*-estimator combined with the *R*-estimator, such as the median, Wilcoxon or Ansari-Bradley-Siegel-Tukey estimator. We demonstrated that the robust properties of the RM-estimators exceed the robust properties of the base *R*- and *M*- estimators for the speckle noise suppression [9]. The RM-estimator used in the proposed 3D filtering scheme is presented as [9, 10]:

$$\theta_{\text{medM}}=\text{MED}\left\{X_p\varphi\left(X_p-\text{MED}\{\bar{\mathbf{X}}\}\right),\ p=1,\ldots,N\right\} \tag{4}$$

where θ_{MedM} is the Median M-type estimator, X_p are data samples, $p=1,\ldots,N$, φ is the normalized function ψ: $\psi(X)=X\varphi(X)$, and $\bar{\mathrm{X}}$ is the primary data sample.

The RM L-filter has been designed by use the combined RM-estimator (4) [9, 10] to increase the robustness of the *L*-filter [11]. The detail description of such a filtering scheme is presented in [4], and in here we proposed its modifications for 3D imaging purposes. So, the 3D RM-L (Rank M-type L) filter is defined in the following way:

$$\theta_{RM\text{-}L}(i,j,k)=\frac{\mathrm{MED}\left\{a_p\left[X_p\psi\left(X_p-\mathrm{MED}\{\bar{\mathrm{X}}\}\right)\right]\right\}}{a_{\mathrm{MED}}} \tag{5}$$

where $X_p\cdot\psi(X_p-\mathrm{MED}\{\bar{\mathrm{X}}\})$ are the selected pixels in accordance with the influence function into a rectangular 3D grid of voxels, $a_p=\int_{p-1/n}^{p/n}h(\lambda)d\lambda\Big/\int_0^1 h(\lambda)d\lambda$ are the weighted coefficients where $h(\lambda)$ is a probability density function [11], a_{MED} is the median of coefficients a_p, the filtering 3D grid size is $N_1\times N_2\times N_3$, $N_p=(2L+1)^2$ and $l_p,m_p,n_p=-L,\ldots,L$, and X_p is the input data sample from the $x(i,j,k)$ of the 3D image contaminated by noise in the rectangular 3D grid where i and j are the 2D spatial axes and k is the time axis (or third dimension). We use the Tukey biweight [12] influence function $\psi_{\text{bi}(r)}(X)=\begin{cases}X^2\left(r^2-X^2\right), |X|\le r\\ 0, |X|>r\end{cases}$ in the proposed 3D RM-L filter.

To improve the properties of impulsive noise suppression of the proposed filter we introduced an impulsive detector, this detector chooses that voxel is or not filtered. The impulsive detector is defined as [13]:

$$\left[\left(rank(X_{ijk})\le s\right)\vee\left(rank(X_{ijk})\ge N_p-s\right)\right]\wedge\left|X_{ijk}-\mathrm{MED}(\bar{\mathrm{X}})\right|\ge U_2 \tag{6}$$

where X_{ijk} is the central voxel in the 3D grid, $s>0$ and $U_2\ge 0$ are thresholds.

The weighted coefficients of the 3D RM L-filter were found using the exponential, Laplacian, and Uniform distribution functions [11, 12]. We note that the coefficients are calculated by each sliding filter window due that the influence function selects whose pixels are used and then compute the weighted coefficients of *L*-filter according with the number of pixels used into the filtering window.

The parameters that characterize the 3D RM L-filter were found after numerous simulations by means of use a 3x3x3 grid (i.e., $N_1\times N_2\times N_3=27$, $l,m,n=-1,\ldots,1$, and $N_p=(2L+1)^2=9$). The idea was to find the parameters values when the criteria PSNR and MAE should be optimum. The optimal parameters of proposed filters are: $s=3$ and $U_2=15$ for the impulsive detector, and $r=15$ for Tukey influence function. The times can change when we use other values for the

parameters, increasing or decreasing the times but the PSNR and MAE values change within the range of ±(5-10)%, it is due that we propose to fix the parameters to can realize the real-time implementation of the 3D RM L-filters.

4 Experimental Results

The described 3D RM L-filter with Tukey biweight influence function and different distribution functions has been evaluated, and its performance has been compared with different nonlinear 2D filters which were adapted to 3D. The filters used as comparative ones were the *modified α-Trimmed Mean* [7, 14], *Ranked-Order* (RO) [15], *Multistage Median* (MSM1 to MSM6) [16], *Comparison and Selection* (CS) [15], *MaxMed* [17], *Selection Average* (SelAve) [15], *Selection Median* (SelMed) [15], *Lower-Upper-Middle* (LUM, LUM Sharp, and LUM Smooth) [18], and *Rank M-type K-nearest Neigbour* (RM-KNN) [3] filters. These filters were computed according with their references and were adapted to 3D imaging. Several experiments were realized to investigate the performances of the different techniques in 3D imaging.

The criteria used to compare the performance of noise suppression of different filters was the *peak signal to noise ratio* (PSNR) [19, 20],

$$\mathrm{PSNR} = 10 \cdot \log\left[\frac{(255)^2}{\mathrm{MSE}}\right], \mathrm{dB} \tag{7}$$

and for the evaluation of fine detail preservation the *mean absolute error* (MAE) was used [19, 20],

$$\mathrm{MAE}=\frac{1}{N_1 N_2 N_3}\sum_{i=0}^{N_1-1}\sum_{j=0}^{N_2-1}\sum_{k=0}^{N_3-1}\left|S(i,j,k)-\hat{f}(i,j,k)\right| \tag{8}$$

where $\mathrm{MSE}=\frac{1}{N_1 N_2 N_3}\sum_{i=0}^{N_1-1}\sum_{j=0}^{N_2-1}\sum_{k=0}^{N_3-1}\left[S(i,j,k)-\hat{f}(i,j,k)\right]^2$ is the *mean square error*, $S(i,j,k)$ is the original free noise 3D image, $\hat{f}(i,j,k)$ is the restored 3D image, and N_1, N_2, N_3 are the sizes of the 3D image.

The experimental ultrasound system used here works in the following way: the microcontroller Microchip PIC 16F84 sends a sign to the motor, so that of a step, sequentially the microcontroller sends a signals to the capture card to acquire the image in the BMP file image format from the ultrasound equipment of General Electric Comp, and in that moment stores it into the computer memory. The rail is the reference point, so, this allows calculating the measure of the explored human organ. There is a distance of 0.069 cm between each captured 2D image. The rail helps obtaining the longitudinal distance and other measures of the ultrasound images (height and width) of the explored organ. Using the presented system it is possible to reconstruct the human organ as an object into 3D space with their real measures. The coor-

dinate z represents each an image 2D of the sweeping in the 3D space, and the coordinates x and y represent the height and width of the 2D image, respectively. Having the 3D image, one can carry out courts in the planes yz, xy, or xz. The proposed ultrasound system is depicted in Figure 1.

The runtime analysis of the 3D RM L-filters and other concerned filters were implemented by using the Texas Instruments DSP TMS320C6711 [5]. This DSP has a performance of up to 900 MFLOPS at a clock rate of 150 MHz [5]. The filtering algorithms were implemented in C language using the BORLANDC 3.1 for all routines, data structure processing and low level I/O operations. Then, we compiled and executed these programs in the DSP TMS320C6711 applying the Code Composer Studio 2.0 [6]. The processing time in seconds includes the time to acquisition, processing, and storing data.

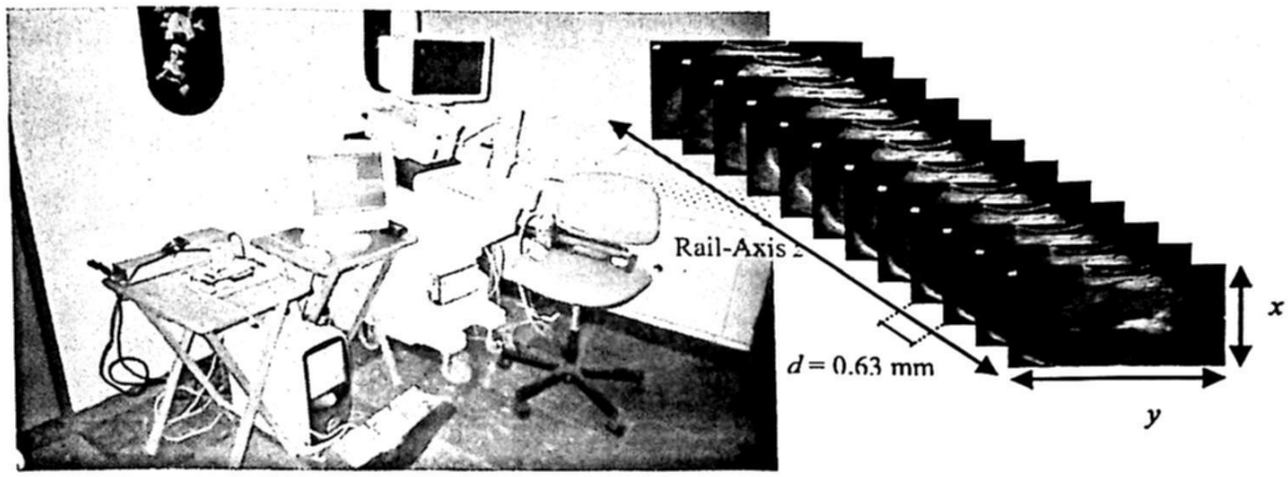


Figure 1. The proposed ultrasound system used to capture the 3D images

The experiment 1 was realized by degraded an ultrasound sequence of 640x480 pixels with 90 frames (3D image of 640x480x90 voxels) with 0.05 and 0.1 of variance of speckle noise added to the natural speckle noise of the sequence. The performance results are depicted in Table 1 by use a frame from xy plane of the sequence. From this table one can see that the 3D RM L-filters provide the best results in comparison to other filters proposed as comparative.

Figure 2 exhibits the visual results of restored images obtained by the use of different filters according to Table 1. In this Figure we observe that the proposed filters provide the better results in speckle noise suppression and detail preservation in comparison with other filters proposed in the literature.

In the experiment 2 we used different voxels cube configurations to provide better noise suppression [7 8]. Figure 3 presents nine configurations of voxels used in the proposed 3D filtering algorithm. It is obvious that by use of less voxels in the different cube configurations the processing time can be decreased. In this experiment the ultrasound sequence was degraded with 20% of impulsive noise. Then, we implemented different cube configurations in the α-Trimmed Mean, MM-KNN, and RM L filters.

Table 2 presents the performance results of different filters in the case of use different cube configurations in the xy plane of the sequence. We observe from this Table that the MM-KNN and α-Trimmed Mean filters provide better results in terms

of PSNR in comparison with the RM L-filter but in the MAE performance the proposed filter provides the better results. About the time to process the algorithms, the proposed RM L-filter has less processing time in comparison with the MM-KNN filter.

Figure 4 shows the visual results obtained by RM L-Filter with the use of different cube configurations in a frame of ultrasound sequence degraded with 20% of impulsive noise according with Table 2. From Figure 4 we observe that the restored images appear to have a good subjective quality.

From the results presented in this paper we notice that the proposed filters can suppress the speckle noise with detail preservation better than other filters proposed in the literature. In the case of impulsive noise suppression the proposed filters have good performance in comparison with other filters.

Finally, the processing time of RM L-filters is acceptable to process 3D images in real time applications because the proposed filters can process QCIF video format with standard film velocity for computer vision systems.

Table 1. Performance results in a frame of ultrasound sequence degraded with speckle noise.

3-D Filters	Speckle noise variance			
	0.05		0.1	
	PSNR	MAE	PSNR	MAE
CS	15.435	32.875	13.843	39.778
LUM Smooth	17.915	25.142	15.440	33.823
LUM Sharp	15.625	30.927	14.444	36.425
LUM	15.518	31.427	14.379	36.748
MaxMed	18.562	24.206	15.919	32.913
MM-KNN CUT	21.554	15.199	18.949	20.995
MM-KNN HAMPEL	21.572	15.169	19.040	20.798
MM-KNN SINE	21.399	14.614	18.640	20.226
MM-KNN BERNOULLI	22.658	13.309	20.075	17.819
MM-KNN TUKEY	22.499	13.446	19.855	18.125
Modified α-Trimmed Mean	20.418	15.124	19.095	18.663
MSM1	20.568	17.624	18.061	23.684
MSM2	20.484	17.789	18.038	23.725
MSM3	22.421	14.206	20.261	18.456
MSM4	21.697	15.401	19.348	20.351
MSM5	19.554	20.207	16.964	27.444
MSM6	22.083	14.688	19.744	19.374
Ranked Order	21.587	14.520	19.802	18.179
SelAve	21.182	17.647	19.192	22.814
SelMed	20.836	15.750	19.013	20.094
RM-L TUKEY UNIFORM	29.876	5.016	28.6175	5.7429
RM-L TUKEY LAPLACIAN	28.797	5.646	28.188	6.0194
RM-L TUKEY EXPONENTIAL	28.034	6.261	26.299	7.6657

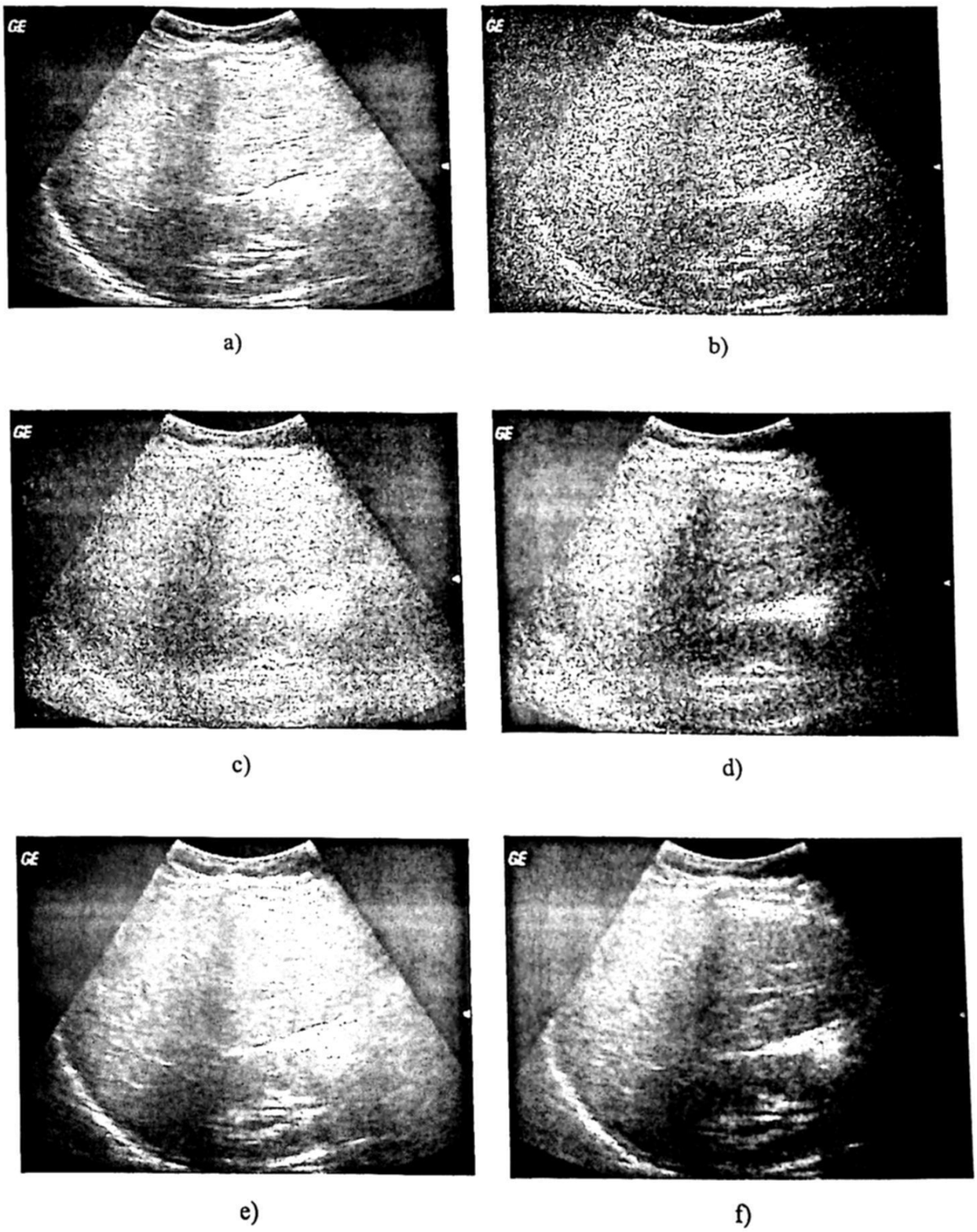


Figure 2. Visual results in a frame of ultrasound sequence. a) original frame, b) frame degraded by 0.05 of variance of speckle noise, c) restored frame by MSM3 filter, d) restored frame by MM-KNN (Bernoulli) filter, e) restored frame by RM L-filter (Uniform), f) restored frame by RM-L filter (Laplacian).

Table 2. Performance results by use different cube configurations in a frame of ultrasound sequence degraded with impulsive noise.

Voxel Configuration on different filters	20% of impulsive noise					
	MM-KNN filter			Modified α-trimmed mean filter		
	PSNR	MAE	Time	PSNR	MAE	Time
a	28.408	4.538	1.6425	26.315	6.979	0.6398
b	29.410	4.415	1.9082	28.236	5.687	0.7127
c	28.768	5.284	4.8228	28.748	5.486	0.8267
d	28.855	5.156	5.1989	28.876	5.348	0.8269
e	28.709	5.289	4.8159	28.680	5.494	0.8267
f	28.683	5.297	4.8297	28.658	5.502	0.8268
g	28.431	5.233	10.0552	28.295	5.684	1.3775
h	28.192	5.384	10.0775	28.037	5.851	1.3769
i	27.919	5.136	20.6575	25.745	7.764	2.1716

	RM L-filter Uniform		
	PSNR	MAE	Time
a	26.831	5.812	1.1485
b	27.670	5.003	1.1627
c	27.572	5.062	2.3251
d	28.295	4.305	2.3247
e	27.532	5.104	2.3289
f	27.541	5.114	2.3254
g	28.068	4.548	3.4934
h	27.438	5.211	3.4993
i	27.768	4.848	4.7732

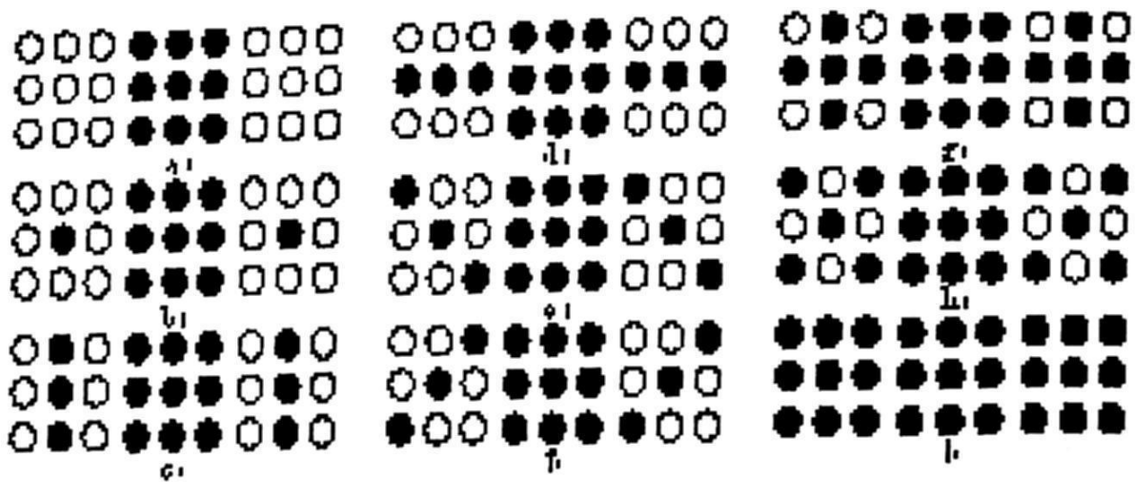


Figure 3. Different configurations of processing cube.

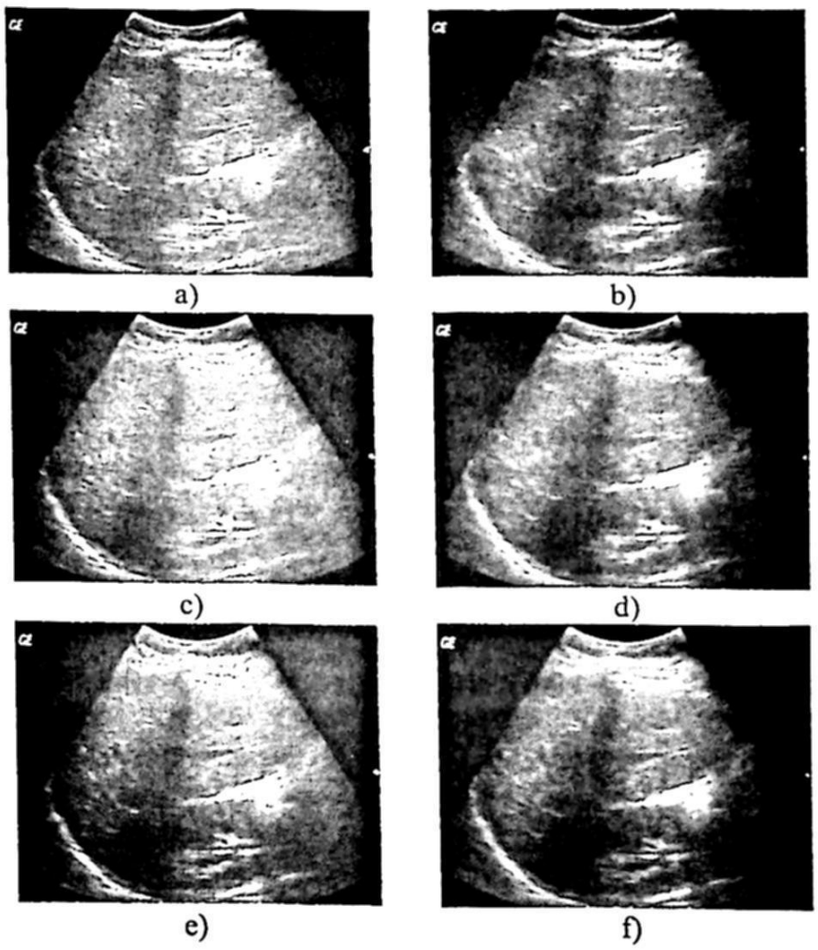


Figure 4. Visual results obtained by RM L-Filter with the use of different cube configurations in a frame of ultrasound sequence degraded with 20 % of impulsive noise. a) original frame, b) restored frame by b cube, c) restored frame by d cube, d) re-stored frame by f cube, e) restored frame by g cube, f) restored frame by i cube.

5 Conclusions

We present the real-time implementation of the 3D RM L-filter for suppression of speckle noise with good detail preservation by means of use of DSP TMS320C6711. The simulation results have demonstrated that the proposed filter consistently outperforms other filters by balancing the tradeoff between speckle noise suppression, detail preservation, and processing time. The proposed filter potentially provides a real-time solution to quality video transmission. The use of the linear combinations of order statistics with the RM-estimator provide to proposed 3D RM L-filter better performance in terms of speckle noise in comparison with the 3D RM-KNN filtering algorithm. Therefore, we realized simulation results in the case of impulsive noise and we notice that the proposed filter provide good results in comparison with different filters.

Acknowledgements. The authors thank the National Polytechnic Institute of Mexico for its financial support.

References

1. Webb, A. G.: Introduction to Biomedical Imaging. Wiley-IEEE Press, Hoboken New Jersey (2002)
2. Abd-Elmoniem, K. Z., Youssef, A. M., Kadah, Y. M.: Real-Time speckle reduction and coherence enhancement in ultrasound imaging via nonlinear anisotropic diffusion. IEEE Trans. Biomed. Eng. 49(9) (2002) 997-1014
3. Ponomaryov, V., Gallegos-Funes, F., Sansores-Pech, R., Sadovnychiy, S.: Real-time Noise suppression in 3D ultrasound Imaging based on Order Statistics. IEE Electronics Letters. 42(2) (2006) 80-82
4. Varela-Benitez, J. L., Gallegos-Funes, F. J., Ponomaryov, V.: RM L-filters for Real Time Imaging. Proc. IEEE 15th International Conference on Computing. (2006) 43-48
5. Texas Instruments: TMS320C6711,TMS320C6711B, TMS320C6711C Floating-Point Digital Signal Processors, SPRS088H. Texas Instruments Incorporated, Dallas (2003)
6. Texas Instruments: TMS320C6000 Code Composer Studio Tutorial, SPRU301C. Texas Instruments Incorporated, Dallas (2000)
7. Nikolaidis, N., Pitas, I.: 3-D Image processing algorithms. John Wiley & Sons, New York (2000)
8. Kim, J. S., Park, H. W.: Adaptive 3-D median filtering for restoration of an image sequence corrupted by impulsive noise. Signal Processing: Image Communication, 16 (2001) 657-668
9. Gallegos-Funes, F. J., Ponomaryov, V. I.: Real-time image filtering scheme based on robust estimators in presence of impulsive noise. Real Time Imaging. 8(2) (2004) 78-90
10. Gallegos-Funes, F., Ponomaryov, V., De-La-Rosa, J.: ABST M-type K-nearest neighbor (ABSTM-KNN) for image denoising. IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences. E88-A(3) (2005) 798-799
11. Pitas, I., Venetsanopoulos, A. N.: Nonlinear Digital Filters: Principles and Applications. Kluwer Academic Publisher, Boston (1990)
12. Hampel, F. R., Ronchetti, E. M., Rouseew, P. J., Stahel, W. A.: Robust Statistics. The approach based on influence function. Wiley, New York (1986)
13. Aizenberg, I., Astola, J., Bregin, T., Butakoff, C., Egiazarian, K., Paily, D.: Detectors of the Impulsive Noise and new Effective Filters for the Impulsive Noise Reduction. Proc. SPIE Image Processing, Algorithms and Systems II. Vol. 5014 (2003) 419-428
14. Bednar, J. B., Watt, T. L.: Alpha-trimmed means and their relationship to median filters. IEEE Trans. Acoust., Speech, and Signal Process. ASSP-32 (1984) 145-153
15. Astola, J., Kuosmanen, P.: Fundamentals of Nonlinear Digital Filtering. CRC Press, Boca Raton-New York (1997)
16. Arce, G. R.: Multistage order statistic filters for image sequence processing. IEEE Trans. Signal Process. 39(5) (1991) 1146-1163
17. Nieminen, A., Neuvo, Y.: Comments of theoretical analysis of the max/median filter. IEEE Trans. Acoust., Speech, and Signal Process. ASSP-36 (1988) 826-827
18. Hardie, R. C., Boncelet, C. G.: LUM filters: a class of rank order based filters for smoothing and sharpening. IEEE Trans. Signal Process. 41 (1993) 1061-1076
19. Bovik, A.: Handbook of Image and Video Processing. Academic Press, San Diego CA (2000)
20. Kotropoulos, C., Pitas, I.: Nonlinear Model-Based Image/Video Processing and Analysis. John Wiley & Sons, New York (2001)

Image Block Edge Classification with Block/Subblock Conversion in the Discrete Cosine Transform Domain

Javier Vega-Pineda, José A. Salinas-Terrazas and Rafael Sandoval-Rodríguez

Instituto Tecnológico de Chihuahua, DEPI
Av. Tecnológico 2909, Chihuahua, Chih., C.P. 31310, México
{jvega, rsandova}@itchihuahua.edu.mx

Abstract. We present a method to classify the edge orientation of the blocks in images under the Discrete Cosine Transform (DCT) domain. The method includes a previous stage to join up or split down the original 8x8 blocks of a traditional JPEG compressed image forming DCT blocks with different dimensions. The change of dimensions is totally realized in the DCT domain and includes two approaches: a general spatial scheme for 2^n and non 2^n related DCT block sizes and a fast scheme just for 2^n related DCT blocks. Both approaches are capable of work dividing DCT block into their subblocks and backwards from the subblocks to the block. The edge classifier takes advantage of the block conversion process to compute the belonging class performing as the spatial domain algorithm version. The method is oriented to applications where the feature extraction from compressed domain images is important.

Keywords. Block edge classification, Discrete cosine transform (DCT), Image feature extraction.

1 Introduction

The use of compressed still digital images is increasing day by day to reduce transmission time and storage space. There are several well known formats in which the images may be coded and one of the most used is JPEG [1]. The JPEG Baseline standard is based on the Discrete Cosine Transform (DCT) [2], and numerous image databases are structured with images under this format. Therefore, new direct image processing in compress domain areas are emerging. Among others, watermarking [3], feature extraction [4-5] and error concealment [6]. Those applications may need to work with image blocks of different size to the 8x8 defined by DCT-JPEG, and therefore the conversion of their size is needed.

After the review of several methods to do the DCT block size conversion, we selected and adapted two of them as the first stage of the Block Edge (BE) classification method [7-10]. The block conversion method of Feng and Jiang [7-8] is the most general in terms of block dimensions, bidirectionality, and the number of arithmetic operations is lower than using the traditional IDCT method. The core factor in the method is to solve a set of linear equations leading to a conversion matrix with properties related to the desired block sizes conversion. The alternative method proposed by He et al [9], is based on a closely replica of the fast Discrete Fourier Transform

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 101-111

Received 20/02/07
Accepted 08/04/07
Final version 18/04/07

(DFT) algorithms applied to the DCT to modify the blocks size. This method is bidirectional and reduces the number of arithmetic operations compared to Feng and Jiang approach, but it is applicable only to $2^n \times 2^n$ blocks. The possibility to modify the DCT image block size will permit to do processing on a global or local region.

The BE classification stage defines the edge orientation (0, 45, 90, 135 degrees) or the edge absence according to the maximum value of previously calculated measures. The edge classification is based on the first probabilistic moment (mean) computed for each of the four subblocks contained into the block under analysis. The edge classification method was proposed by Sung and Kang [10] and is defined in both domains spatial and DCT. However, taking advantage of previous results from the block conversion stage we modified the DCT approach implementing a method similar to the spatial approach but maintaining the image in the DCT domain. Li et al [11] analyze a set of DCT coefficients to determine the edge orientation which is less efficient than use the mean of subblocks proposed by [10], and more recently they [12] proposed a new scheme based on the Haar transform with a drastic reduction in the number of arithmetic operations but it is a method to be applied in the pixel domain. The modified Sung and Kang approach [10] prove to work very efficiently, and the metrics used to classify the edges based on the mean are very intuitive.

The method proposed in this paper turns to be an integration of several modified algorithms adapted and optimized to provide a two fold image processing application which performs totally in the DCT domain: the classification of the image blocks, and the modification of the relationship between the DCT coefficients of its blocks and their subblocks. Therefore, image global and local feature extraction (edges orientation) in the compress – DCT domain is possible. Experimental results are presented and the general structure of the method explained.

The remainder of this article is organized as follows. In section 2, the basic aspects of the selected block conversion algorithms are summarized. Section 3 explains how we have simplified the selected BE algorithm and the orientation measures explained. Section 4 describes the integrated method as a tool to classify BE using the algorithms explained in sections 2 and 3. Finally, section 5 reports results in terms of image processing and draws some conclusions.

2 Algorithms for Conversion Between DCT Blocks and Subblocks

The Feng and Jiang [7] and also the He at al [9] methods are summarized, both relating the 2-D DCT blocks and their subblocks. Before presenting the two conversion algorithms, we define notation and formulate the specific problem. Given a 2-D block of pixels B, its DCT will be C_B. B can be divided into B_{ij} subblocks where the corresponding DCT for each subblock is C_{ij}. This is illustrated in Fig. 1. In the first approach [7], we will solve a set of linear equations relating the DCT basis used by the block and its sub-blocks. In the second approach [9], a fast algorithm based on DCT-II and DCT-IV which resembles the Fast Fourier Transform structure is used to relate C_B and C_{ij}.

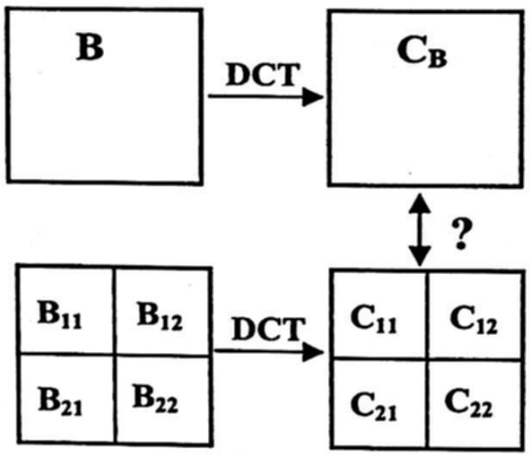


Fig. 1. The relationship between DCT a block and its subblocks

The definition for the 2-D DCT (or DCT-II) used for the algorithms development is given by

$$C_B(u,v) = \sqrt{\frac{4}{R \times C}}\alpha(u)\alpha(v) \cdot \sum_{i=0}^{R-1}\sum_{j=0}^{C-1} x(i,j)\cos\left(\frac{(2i+1)u\pi}{2R}\right)\cos\left(\frac{(2j+1)v\pi}{2C}\right) \tag{1}$$

where $\{x(i,j)\}$ are the elements of B, and $\alpha(u) = \begin{cases} \sqrt{\frac{1}{2}}, & \text{for } u = 0 \\ 1, & \text{otherwise.} \end{cases}$

2.1 Method Based on the Spatial Relationship Between DCT Blocks/Subblocks

The forward conversion will be defined as joining subblocks C_{ij} to reach a block C_B. The problem is reduced to solve the linear relationship between two families of DCT basis functions with equal dimensions. Assuming that C_B has dimensions of R x C = LN x MN, the subblock structure has L x M subblocks, with size each of N x N. The first basis corresponds to the block and the second basis corresponds to one of the subblocks forming the block, i.e., $b_2(k,t)_{MN\times MN} = \cos(k\pi t / MN)$, $k = 0, 1, ..., MN\text{-}1$ and $b_1(k,t)_{N\times N} = \cos(k\pi t / N)$, respectively. $b_2(k,t)$ must have the same size than $b_1(k,t)$, then this last is expanded by zero padding forming $b(k,t)_{MN \times MN}$ a matrix with b_1 submatrices in its main diagonal. The linear relationship is established as

$$b_2(k,t) = \sum_{j=0}^{MN-1} a(k,j)b(j,t). \tag{2}$$

where $t = (2i+1)/2$, ($i = 0,1,...,MN-1$). (3) can be written in compact matrix form as b_2=Ab.

Solving for a unique $a(k,j)_{MN \times MN}$ (A) and after applying the normalization constants $\alpha_n(.)$ [7], will give us the forward conversion matrix A^* between subblocks to one block. A^{*-1} is the backward conversion matrix. A numerical example follows: $N = 3$, $L = 2$ and $M = 2$, then four 3 x 3 subblocks (2 by row and 2 by column) will be converted to one 9 x 9 block. First, compute $A^*=\alpha_n b_2 b^{-1}$, and second, apply A^* to the rows and columns of C_{ij}. The transformation can be expressed as,

$$C_B = \sqrt{\frac{1}{LM}} A^* C_{ij} A^{*T}. \tag{3}$$

The forward conversion matrix A* is computed as

$$A^* = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0.9107 & 0.4082 & -0.0632 & -0.9107 & 0.4082 & 0.0632 \\ 0 & 1 & 0 & 0 & -1 & 0 \\ -0.3333 & 0.8165 & 0.4717 & 0.3333 & 0.8165 & -0.4717 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0.244 & -0.4082 & 0.8797 & -0.244 & -0.4082 & -0.8797 \end{bmatrix},$$

with an input set of subblocks C_{ij}, results in a block C_B:

$$C_{ij} = \begin{bmatrix} \begin{bmatrix} 167 & 2 & 0 \\ -4 & -1 & 0 \\ 2 & -1 & 0 \end{bmatrix} & \begin{bmatrix} 176 & -12 & 5 \\ -18 & 8 & 1 \\ 5 & -1 & -1 \end{bmatrix} \\ \begin{bmatrix} 192 & 3 & -1 \\ -10 & -3 & 0 \\ 0 & 1 & 1 \end{bmatrix} & \begin{bmatrix} 204 & -20 & 3 \\ 18 & -5 & -2 \\ -3 & 1 & -3 \end{bmatrix} \end{bmatrix} \quad C_B = \begin{bmatrix} 370 & -15 & 18 & -10 & 3 & -1 \\ -27 & 0 & -6 & 3 & 1 & -2 \\ -15 & 22 & -5 & -2 & 2 & 1 \\ 5 & -8 & -1 & 0 & -1 & -1 \\ 2 & 0 & 0 & 1 & -2 & 2 \\ 1 & 0 & 0 & -2 & 1 & -1 \end{bmatrix}.$$

Solving (3) for C_{ij} will define the block to subblocks transformation. The algorithm is valid for blocks with dimensions equal and different to 2^n x 2^n, and also there is a definition for 1-D vectors.

2.2 Method Based on the Fast Computation Between DCT Blocks/Subblocks

This method is mainly based on the structure of decimation-in-frequency FFT algorithm [9]. Mapping the structure to the DCT, a transformation matrix T_N can be decomposed into a straight forward combination of the submatrix $T_{(N/2)}$, where N is the number of values to transform. The conversion is applied only for 2^n x 2^n blocks. For 2-D DCT forward subblocks to block conversion, the related equations are given as

$$C_B = \frac{1}{2} T_N \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} T_N^T = \frac{1}{2} P_N \begin{bmatrix} Y_1 & Y_2 \\ Y_3 & Y_4 \end{bmatrix} P_N^T, \tag{4}$$

$$\begin{aligned} Y_1 &= C_{11} + S_{\frac{N}{2}} C_{21} + C_{12} S_{\frac{N}{2}} + S_{\frac{N}{2}} C_{22} S_{\frac{N}{2}} \\ Y_2 &= \left(C_{11} + S_{\frac{N}{2}} C_{21} - C_{12} S_{\frac{N}{2}} - S_{\frac{N}{2}} C_{22} S_{\frac{N}{2}} \right) B_4 \\ Y_3 &= B_4^T \left(C_{11} + C_{12} S_{\frac{N}{2}} - S_{\frac{N}{2}} C_{21} - S_{\frac{N}{2}} C_{22} S_{\frac{N}{2}} \right) \\ Y_4 &= B_4^T \left(C_{11} - C_{12} S_{\frac{N}{2}} + S_{\frac{N}{2}} C_{22} S_{\frac{N}{2}} - S_{\frac{N}{2}} C_{21} \right) B_4 \end{aligned} \tag{5}$$

where C_{ii} = 2-D DCT(B_{ii}), T_N is the N-values DCT transformation matrix, P_N is the N x N permutation matrix (rows in order 1, 3,..., N-1, 2, 4,..., N), $S_{N/2} = T_{N/2} \overline{I_{N/2}} T_{N/2}^T$, $\overline{I_N}$ is the reverse identity matrix, $B_4 = T_{N/2} \left(T_{N/2}^{IV} \right)^T \overline{I_{N/2}}$, $T_{N/2}^{IV}$ is the $N/2$-values DCT-IV (see Appendix A) and Y_i are linear combinations of the four transform matrices (DCT sections).

The linear combinations for 2-D DCT backward block to subblocks conversion are given as

$$\begin{aligned} C_{11} &= \frac{1}{4} (K_1 + K_2 + K_3 + K_4) \\ C_{12} &= \frac{1}{4} (K_1 - K_2 + K_3 - K_4) S_{\frac{N}{2}} \\ C_{21} &= \frac{1}{4} S_{\frac{N}{2}} (K_1 + K_2 - K_3 - K_4) \\ C_{22} &= \frac{1}{4} S_{\frac{N}{2}} (K_1 - K_2 - K_3 + K_4) S_{\frac{N}{2}}, \end{aligned} \tag{6}$$

where the $(N/2)$x$(N/2)$ matrices K_i are given as

$$K_1 = Y_1, \quad K_2 = Y_2 B_4^T, \quad K_3 = B_4 Y_3, \quad K_4 = B_4 Y_4 B_4^T. \tag{7}$$

A numerical example follows: converting a 4 x 4 block into its four 2 x 2 subblocks $N = 4$ (backward conversion). Solving (4) for Y_i's and then substituting in (6) through (7) for K_i's:

$$C_B = \begin{bmatrix} 218 & 45.63 & 22 & -4.8 \\ 74.7 & -66.56 & -24.14 & -26.8 \\ -58.5 & 30.76 & 0.5 & -13.29 \\ -3.89 & -33.3 & -7.32 & 14.06 \end{bmatrix} \quad P_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \begin{bmatrix} Y_1 & Y_2 \\ Y_3 & Y_4 \end{bmatrix} = \begin{bmatrix} 436 & 44 & -9.66 & 91.25 \\ -117 & 1 & -26.56 & 61.52 \\ -7.76 & -14.64 & 28.11 & -66.6 \\ 149.40 & -48.28 & -53.6 & -133.11 \end{bmatrix}$$

$$T_2 = \begin{bmatrix} 0.7071 & 0.7071 \\ 0.7071 & -0.7071 \end{bmatrix} \quad S_2 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \quad T_2^{IV} = \begin{bmatrix} 0.9239 & 0.3827 \\ 0.3827 & -9239 \end{bmatrix} \quad B_4 = \begin{bmatrix} -0.383 & 0.924 \\ 0.924 & 0.383 \end{bmatrix}$$

$$\begin{bmatrix} K_1 & K_2 \\ K_3 & K_4 \end{bmatrix} = \begin{bmatrix} 436 & 44 & 88 & 26 \\ -117 & 1 & 67 & -1 \\ 141 & -39 & -67 & -93 \\ 50 & -32 & -106 & -38 \end{bmatrix} \quad \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} = \begin{bmatrix} 149.5 & -15.5 & 139 & -18 \\ -26.5 & -17.5 & -7 & -2 \\ 112.5 & 50.5 & 35 & 9 \\ -1.5 & -17.5 & 85 & -1 \end{bmatrix}$$

In this method most of the matrices can be computed before the block conversion. Even when the reordering operations have not computational cost [9], they are required to arrive to the correct values position. There is also a 1-D version of the algorithm.

3 Compress Domain Block Edge (BE) Classification Method

The BE classification method in the pixel domain working on an image can be easily described as follows:

i. The image $N \times N$ blocks are divided in their symmetrical four subblocks.
ii. The mean (first probabilistic moment) is computed for each subblock: S_{11}, S_{12}, S_{21}, S_{22}.
iii. Define the threshold value for edge or not edge block (heuristically), d_{NE}.
iv. The measures for directional edge pattern are computed using equations in Table 1.
v. The measure with the maximum value including the edge threshold value, defines the edge class.

Table 1. Measures for block directional patterns

Edge Direction *(radians)*	*Measure*
No Edge	d_{NE} (set by user)
0	$d_0 = \frac{1}{2}\lvert S_{11} + S_{12} - (S_{21} + S_{22})\rvert$
$\pi/4$	$d_{45} = \max\left\{\frac{1}{3}\lvert 3S_{11} - (S_{12} + S_{21} + S_{22})\rvert, \frac{1}{3}\lvert 3S_{22} - (S_{11} + S_{12} + S_{21})\rvert\right\}$
$\pi/2$	$d_{90} = \frac{1}{2}\lvert S_{11} + S_{21} - (S_{12} + S_{22})\rvert$
$3\pi/4$	$d_{135} = \max\left\{\frac{1}{3}\lvert 3S_{12} - (S_{11} + S_{21} + S_{22})\rvert, \frac{1}{3}\lvert 3S_{21} - (S_{11} + S_{12} + S_{22})\rvert\right\}$

The computation of the measures in the DCT domain also is defined in [10]. However, because the directionality measures are based only on the average subblock values, we proposed a modification to the method. Is well known that the DC coeffi-

cient in a DCT block is N times the whole block average value [13], this can be written as

$$\text{(In pixel domain): } m = \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} x(i,j)\ ;\ \text{(In DCT domain): } m = \frac{1}{N} C_B(0,0)\,. \qquad (8)$$

Therefore, using the DCT-DC coefficient from each subblock inside the block the five measures at Table 1 can be computed and a BE class defined. The DCT subblocks generation may then be performed by any of the two block/subblock conversion methods described in section 2.

4 Image Feature Extraction Tool: Integrating Block/Subblock Conversion and BE Classification

In Fig. 2, the methods explained in sections 2 and 3 are integrated as an image global or local region feature extraction tool. The DCT blocks or subblocks are converted to the desired size according to the final application using the tool. The converted blocks are 2-D subsampled (ND-ND) leaving the DC coefficients to compute the BE directional measures. The measure with the maximum value defines a pointer to the edge class and the corresponding tag is assigned to the block.

An important difference between the methods for block conversion presented at section 2 is the scope obtained in the size conversion. The fast computation method (section 2b), works in a recursive structure, the previous 2^n x 2^n subblocks are required to the next conversion, i.e., the conversion from 8^2 to 2^2 blocks needs to compute previously the 8^2 to 4^2 conversion.

With the spatial relationship method (section 2a), an A* matrix can be previously calculated and the conversion 8^2 to 2^2 be performed in one step. Be A* the N^2 to $(N/2)^2$ conversion matrix and be D^* the $(N/2)^2$ to $(N/4)^2$ conversion matrix. Notice that A* is an N x N matrix and D^* is (N/2) x (N/2) matrix. Therefore D^* is resized to N x N by zero padding out of the main block diagonal. Solving (3) for C_{ij}, the N^2 to $(N/2)^2$ conversion is defined as

$$C_{ij_{(N/2)^2}} = \sqrt{LM}\, A^{*-1} C_{B(N^2)} \left(A^{*-1}\right)^T , \qquad (9)$$

zero padded D^* is applied to (9),

$$C_{ij_{(N/4)^2}} = LM \cdot D_Z^{*-1} A^{*-1} C_{B(N^2)} \left(A^{*-1}\right)^T \left(D_Z^{*-1}\right)^T , \text{where } D_Z^* = \begin{bmatrix} D^* & 0 \\ 0 & D^* \end{bmatrix}_{N^2} . \qquad (10)$$

The 8^2 to 2^2 conversion matrix is then computed as $A_{8-2}^* = D_Z^{*-1} A^{*-1} = \left(A^* D_Z^*\right)^{-1}$.

Using the fast DCT approach there is an important reduction on the computational cost [9]. Furthermore, the position of the subblocks DC coefficients in (6) remains the same before and after the multiplications of the linear combinations by $S_{N/2}$, therefore,

in our application the complexity is reduced avoiding such multiplications. Results applying the image feature extraction tool are presented in the next section.

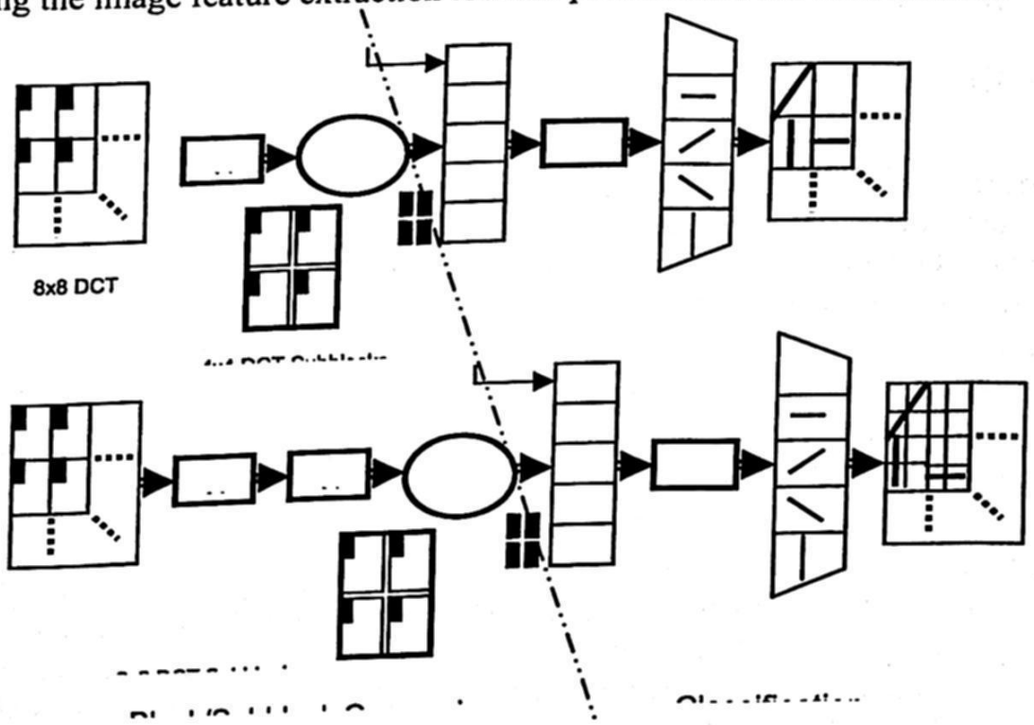


Fig. 2. Integrating Block/Subblock Conversion and BE Classification

5 Experimental Results and Conclusions

The results of analyzing Barbara image is presented at Fig. 3, for threshold values d_{NE} of 10 and 20, and for 8^2 and 4^2 blocks with 4^2 and 2^2 subblocks respectively [14]. Those blocks tagged as an oriented edge where substituted by its corresponding edge image, and those blocks below the d_{NE} value where left blanked. The classified edges where assembled as an image to compare the results with the original Barbara. For small threshold value d_{NE} much more blocks are classified as BE, this can be easily detected at the head and the table cloth of Barbara. The change of block size from 8^2 to 4^2 as was expected, increases the image details and therefore the number of blocks classified as BE. A set of images with the detail of Barbara face are show in Fig. 4, supporting the previous comment.

The BE Classification Method proposed in [10] is defined to work in the DCT domain with blocks of size 8^2 or smaller. With the spatial-like approach proposed in this article it is possible to apply the method to blocks of dimensions above of 8^2, because just the four subblocks DC coefficients are needed to compute the measures for direc tional patterns. In order to compute the BE pattern in a 16^2 block, the DC coefficients of its four corresponding 8^2 subblocks are needed.

The integration of both stages, the block/subblocks conversion and the BE classification all in the DCT domain performed successfully. The modifications to the block/subblocks methods simplified the whole tool complexity keeping the flexibility of choice between the general [7] and the fast DCT computation schemes [9].

Fig. 3. Barbara images BE classified. Top left, Original. Top center and right, 8^2 blocks, d_{NE} = 20 and d_{NE} = 10, respectively. Bottom left and right, 4^2 blocks, d_{NE} = 20 and d_{NE} = 10

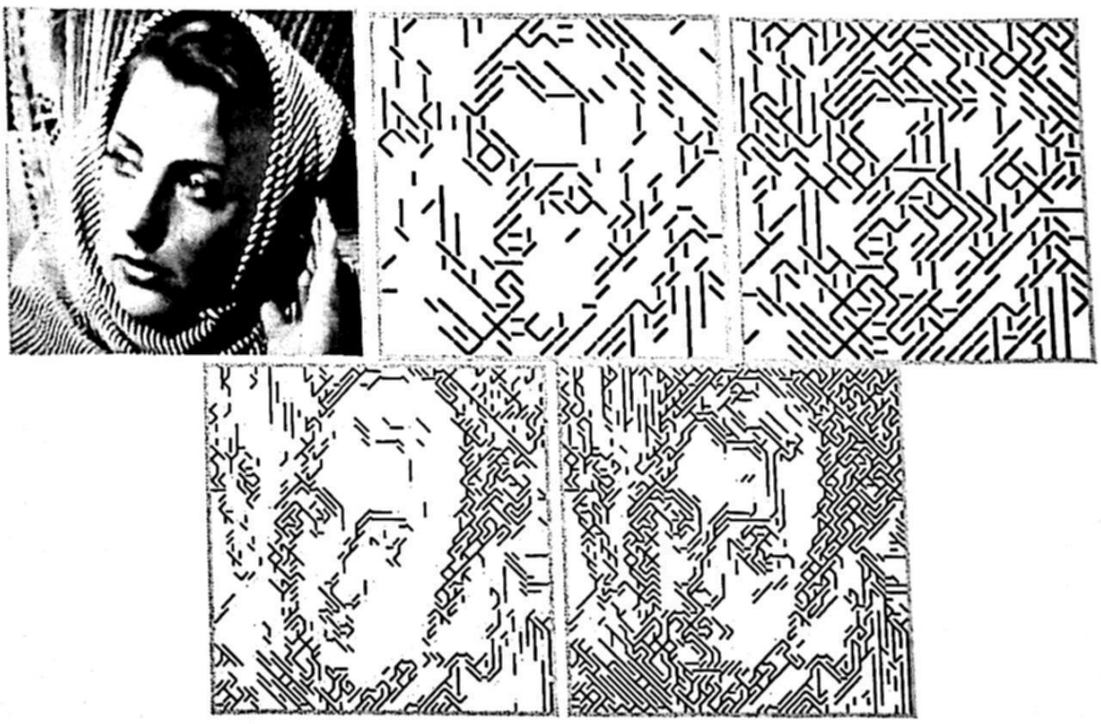


Fig. 4. Barbara face images BE classified. Top left, Original. Top center and right, 8^2 blocks, d_{NE} = 20 and d_{NE} = 10, respectively. Bottom left and right, 4^2 blocks, d_{NE} = 20 and d_{NE} = 10

The BE classifier resembling closely the pixel scheme [10] in place of the DCT version, takes advantage of the preprocessing DCT blocks stages simplifying even further the edge classification. Pattern recognition [14] and video analysis [10] in the DCT domain are other research directions for the work here presented. The digital image processing area in the compress domain is an emerging and interesting area, this article is a small example.

Acknowledgments. The authors are grateful to the support given by the Dirección General de Institutos Tecnológicos (DGEST), the CONACYT, and the Instituto Tecnológico de Chihuahua.

References

1. Pennebaker W.E., Mitchell J.L.: JPEG: Still Image Data Compression Standard. Kluwer Academic Publishers, Norwell, MA, (1992)
2. Rao K.R., Yip P.: Discrete Cosine Transform: Algorithms, Advantages, Applications. Academic Press, San Diego, CA (1990)
3. Wen-Nung L., Guo-Chiang L., Chih-Liang W., Ta-Chun W.: Robust Image Watermarking On the DCT Domain, Proc. IEEE International Symposium On Circuits and Systems, Vol. I. Geneva, Switzerland, May 28-31 (2000) 228-231
4. Jiang J., Armstrong A., Feng G.C.: Direct Content Access and Extraction form JPEG Compressed Images, Pattern Recognition, Vol. 35. Elsevier Science Ltd (2002) 2511-2519
5. Beetnara B., Sun W.Y., Yong M.R.: Fast MPEG-7 Visual Description Extraction Using DCT Coefficient, Proc. IEEE Conference of Convergence Technologies for Asia-Pacific Region, Vol. 3. Bangalore, India, Oct. 15-17, (2003) 1136-1139
6. Hemami S.S., Gray R.M.: Subband-Coded Image Reconstruction for Lossy Packet Network, IEEE Trans. on Signal Processing, Vol. 6. No. 8. (1999) 202-204
7. Jiang J., Feng G.: The Spatial Relationship of DCT Coefficients Between a Block and Its Sub-blocks, IEEE Trans. on Signal Processing, Vol. 50. No. 5. (2002) 1160-1169
8. Jiang J., Feng G.: Image Spatial Transformation in DCT Domain, Proc. IEEE International Conference on Image Processing, Vol. 3. Thessaloniki, Greece, Oct 7-10 (2001) 836-839
9. He Z., Bystrom M., Nawab H.: Bidirectional Conversion Between DCT Coefficients of Blocks and Their Subblocks, IEEE Trans. on Signal Processing, Vol. 53. No. 8. (2005) 2835-2841
10. Sung C.H., Kang K.: A Compressed Domain Scheme for Classifying Block Edge Patterns, IEEE Trans. on Signal Processing, Vol. 14. No. 2. (2005) 145-151
11. Li H., Liu G., Li Y.: An Effective Approach to Edge Classification from DCT Domain, Proc. IEEE International Conference on Image Processing, Vol. 1. Rochester, NY, Sep 22-25 (2002) 940-943
12. Li H., Ngan K.N.: Fast and Efficient Method for Block Edge Classification, Proc. International Conference on Communications and Mobile Computing, ACM Press, New York, NY, July 03-06, (2006) 67-72
13. Liu J.G., Liu Y.Z., Wang G.Y.: Fast DCT-I, DCT-III, and DCT-IV via Moments, Eurasip Journal on Applied Signal Processing, Vol. 12. Hindawi Pub. Co., (2005) 1902-1909
14. Salinas-Terrazas J. A.: Búsqueda de Patrones Geométricos en Imágenes en el Dominio de la Transformada Discreta del Coseno, M. Sc. Thesis, Instituto Tecnológico de Chihuahua, Chihuahua, Chih., México, (2007)

Appendix: A

The Discrete Cosine Transform II and IV are defined by different equations [2]. The 2-D DCT-II is the one in equation (1). The 1-D DCT-IV is defined as

$$C_B^{IV}(k) = \sqrt{\frac{2}{N}} \sum_{n=0}^{N-1} x(n) \cos \frac{\pi(2n+1)(2k+1)}{4N} \tag{11}$$

where $k = 0,...,N\text{-}1$. The $N \times N$ transformation matrix for the DCT-IV is defined as

$$T_N^{IV}(i,j) = \sqrt{\frac{2}{N}} \cos \frac{\pi(2i+1)(2j+1)}{4N}, \quad i,j \in [0,...,N-1]. \tag{12}$$

A New Efficient Approximation Algorithm for Chromatic Number

Guillermo De Ita[1], Meliza Contreras[2] Erica Vera[1]

[1] Universidad Autónoma de Puebla, deita@cs.buap.mx, ee_vera@hotmail.com
[2] Universidad Angelópolis, mel_22281@hotmail.mx

Abstract. We design a new approximation polynomial-time algorithm for the graph coloring problem. Our proposal is based on selecting, in iterative manner, a critical vertex v of the graph. The criterion to select is based on choose the node with maximum degree and with maximum degree of its neighborhood into the set of vertices composing odd cycles. The algorithm consists of two embedded loops. While in the internal loop a critical node is selected to be colored and it is deleted as well as its upon edges from the current graph. The external loop controls when there is not possible to select more vertices and, while remains odd cycles in the current graph, new colors are used. The stop criterion to finish the two loops is when the current subgraph is bipartite.
Our algorithm establishes an average number of $(\sqrt{4\delta+1}+2\delta-1)/2$ colors for approximate the chromatic number of any graph G, δ being the initial average degree of the input graph G.

1 Introduction

The problem to determine the minimum value of colors needed for coloring a graph is a NP-complete problem, even for graphs G with degree (maximum degree) $\Delta(G) \geq 3$. A consequence of this is that there is not a complete theoretical characterization of colorability [8].

The graph coloring problem is an abstraction of certain types of scheduling problems. In the graph k-coloring problem we wish to assign each vertex one of k colors such that every pair of vertices connected with an edge are assigned different colors. The chromatic number of a graph G denoted by $\chi(G)$ is the minimum value k such that G has a k-coloring. This problem arises in a host of applications, and was one of the 22 NP-complete problems on Karp's list. Subsequently, much effort was spent on trying to design efficient approximation algorithms, namely, given a k-colorable graph to try to color it with as few colors as possible [1].

One of the first bound to color a 3-colorable graph was established by Wigderson [13], he showed how to color 3-colorable graphs with at most $3 \cdot \lceil \sqrt{n} \rceil$ colors, where n is the number of nodes on the graph. Blum and Karger [3] applied semidefinite programming (SDP) to improve this bound to $O(n^{3/14})$, where the notation O is used to suppress polylogarithmic factors. Recently, Arora, et. al. [1] using stronger SDPs have improved the bound to $O(n^{0.2111})$.

© S. Torres, I. López, H. Calvo. (Eds.)
Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 113-122

Received 15/02/07
Accepted 08/04/07
Final version 22/04/07

On the other hand, no polynomial-time algorithm is known for coloring any graph where the ratio of the number of colors used to the optimal number is bounded by a constant. In fact, guaranteeing a small constant bound on the ratio is NP-Hard [2]. For example, is known that if there were an approximation polynomial-time algorithm that uses fewer than $(4/3)\chi(G)$ colors, then the 3-colorability problem could be solved in polynomial time and then, $NP = P$.

For unrestricted number of colors we know that a graph coloring can not be approximated with ratio n^{ϵ} for some ϵ; the current value for the exponent ϵ is $1/10$ [4]. However, the upper bound that is achieved by the known approximation algorithm, i.e. the proposal in [1], is still far to the theoretical bound.

We present in this article, a polynomial-time algorithm which uses an average number of $(\sqrt{4\delta+1}+2\delta-1)/2$ colors for approximate $\chi(G)$, δ being the initial average degree of the input graph G.

2 Preliminaries

Let $G = (V, E)$ be an undirected graph with vertex set (or nodes set) V and set of edges E. Two vertices v and w are called *adjacent* if there is an edge $\{v, w\} \in E$, joining them. Sometimes, we denote with $E(G)$ and $V(G)$ rather than E and V to emphasize that these are the edges and vertex sets of a particular graph G. The *Neighborhood* of $x \in V$ is $N(x) = \{y \in V : \{x, y\} \in E\}$ and its *closed neighborhood* is $N(x) \cup \{x\}$ which is denoted by $N[x]$. Note that v is not in $N(v)$.

We denote the cardinality of a set A, by $|A|$. Given a graph $G = (V, E)$, the degree of a vertex $x \in V$, denoted by $\delta(x)$, is $|N(x)|$. The size of the neighborhood of x, $\delta(N(x))$, is $\delta(N(x)) = \sum_{y \in N(x)} \delta(y)$. The maximum degree of G or just the degree of G is $\Delta(G) = max\{\delta(x) : x \in V\}$, while we denote with $\delta_{min}(G) = min\{\delta(x) : x \in V\}$ and with $\delta(G) = (2 \cdot |E|)/|V|$ the average degree of the graph.

Given a subset of vertices $S \subseteq V$ the subgraph of G denoted by $G|S$ has vertex set S and set of edges $E(G|S) = \{\{u, v\} \in E : u, v \in S\}$. To $G|S$ is called the *subgraph of G induced by S*. We write $G - S$ to denote the graph $G|(V - S)$. The subgraph induced by $N(v)$ is denoted as $H(v) = G|N(v)$ which has to $N(v)$ as the set of nodes and all edges upon them.

A path from a vertex v to a vertex w in a graph is a sequence of edges: $v_0v_1, v_1v_2, \ldots, v_{n-1}v_n$ such that $v = v_0$ and $v_n = w$ and v_k is adjacent to v_{k+1} for $0 \leq k < n$ and, the length of the path is n. A simple path is a path such that $v_0, v_1, \ldots, v_{n-1}, v_n$ are all distinct. A cycle is just a nonempty path such that the first and last vertices are identical, and a simple cycle is a cycle in which no vertex is repeated, except that the first and last vertices are identical.

A k-cycle is a cycle of length k, that is, a k-cycle has k edges. A cycle of odd length is called an odd cycle, while a cycle of even length is called an even cycle. A graph G is acyclic if it has not cycles.

A *complete graph* of n nodes has $n(n+1)/2$ distinct edges, we denote K_n the complete graph of n nodes. A graph G is a *regular graph* if all vertices have the same degree, G is $k-$*regular* if it is regular, of degree k.

A connected component of G is a maximal induced subgraph of G, that is, a connected subgraph which is not a proper subgraph of any other connected subgraph of G. Note that, in a connected component, for every pair of its vertices x, y, there is a path from x to y. If an acyclic graph is also connected, then it is called a free tree.

A coloring of a graph $G = (V, E)$ is an assignment of colors to its vertices. A coloring is *proper* if adjacent vertices always have different colors. An k-coloring of G is a mapping from V into the set $\{1, 2, \ldots, m\}$ of k "colors". The chromatic number of G denoted by $\chi(G)$ is the minimum value k such that G has a proper k-coloring. If $\chi(G) = k$, G is then said to be k-chromatic. The problem to determine the value $\chi(G)$ is polynomial computable when $\chi(G) \leq 2$, but when $\chi(G) \geq 3$, the problem becomes NP-complete, even for graphs G with degree $\Delta(G) \geq 3$.

Given a graph $G = (V, E)$, $S \subseteq V$ is an independent set in G if for whatever two vertices v_1, v_2 in S, $\{v_1, v_2\} \notin E$. Let $I(G)$ be the set of all independent sets of G. An independent set $S \in I(G)$ is *maximal* if it is not a subset of any larger independent set and, it is *maximum* if it has the largest size among all independent sets in $I(G)$.

Let $G = (V, E)$ be a graph, G is a *bipartite graph* if V can be partitioned into two subsets U_1 and U_2, called *partite sets*, such that every edge of G joins a vertex of U_1 and a vertex of U_2.

If $G = (V, E)$ is a k-chromatic graph, then it is possible to partition V into k independent sets $V_1, V_2, ..., V_k$, called *color classes*, but it is not possible to partition V into $k - 1$ independent sets.

3 Polynomial Coloring Procedures

The main class of graphs which is known to be colored in polynomial time is the class of bipartite graphs.

Lemma 1 *A graph G has chromatic number 2 if and only if G is bipartite.*

Since the partite set U_1 can be colored with the first color while the other partite set U_2 is coloring with the second color. Furthermore, as the bipartite property can be recognized in polynomial time, then a 2-colorable graph can be recognized in polynomial time, based on the following property.

Lemma 2 *A graph G is bipartite if and only if G contains no odd cycles.*

Given an input connected graph $G = (V, E)$, if we apply the depth-first search over G starting the search, for example with the node $v \in V$ of minimum degree, we obtain a depth-first graph T_G, which we will denote as $T_G = dfs(G)$. If G is an acyclic graph, we call to T_G the spanning tree of G, and in this case, we can color T_G with only two colors. We color the vertices in T_G by levels, in alternating way between the two colors in accordance with the change of level, that is, all vertices in same level have same color.

The depth-first search also allow us to detect the odd cycles of a graph. We present, in a schematic way, a recursive procedure for the depth-first search. We show in the following procedure, the different status of a node during the search. At the beginning, every node is not discovered and, when a node u as well as all its adjacent nodes have been visited then u is marked as finished.

Procedure dfs(v)

1. Mark v as discovered
2. For each node $w \in N(v)$
   (a) If w is not discovered then $dfs(w)$
   (b) else mark the edge $\{w, v\}$ as a *back edge*
3. Mark v as finished
4. Returns

The procedure dfs runs in time $O(m+n)$ where n and m are the number of nodes and the number of edges of the input graph G, respectively. Thus, dfs is a linear-time procedure over the length of G which can be used for coloring a graph.

Each *back edge* that we find during the depth-first search marks the beginning and the end of a base cycle (or fundamental cycle). Let $C = \{C_1, C_2, ..., C_k\}$ be the set of fundamental cycles found during the depth-first search. Each back-edge $c_i \in T_G$ determines a base cycle $C_i \in C$, $i = 1, \ldots, k$.

If T_G is acyclic or contains only even fundamental cycles, then T_G is bipartite and it is colorable with just two colors. Like the case when T_G is a tree, we could advance visiting nodes in T_G by levels assigning to each node of the same level the same color. The two colors are applied in an alternating way in accordance with the change of level.

Given two distinct base cycles C_i and C_j from C, if C_i and C_j share common edges we say that both cycles are *intersected*, that is, $C_i \oplus C_j$ conforms a new cycle, where $\oplus$ denotes the operation or-exclusive between the set of edges of the cycles. If the cycles C_i and C_j have no common nodes nor edges then we say that both cycles are *independent*, that is, $C_i \oplus C_j = C_i \cup C_j$. While if the cycles have not common edges but maybe they have a common node, we say that both cycles are *non-intersected*. Note that a pair of independent cycles are non-intersected.

Also is known, that any simple odd cycle request of three colors to be colored. Then, we can recognize in polynomial time a set of graphs which are 3-colorable. We present in the following section the prerequisites so that a graph is 3-colorable.

3.1 A 3-Coloring Algorithm

Theorem 1 *If T_G, the resulting depth-first graph of an input graph G contains only non-intersected base cycles, then $\chi(G) \leq 3$ and a 3-coloring is done in linear time.*

Proof: we present as proof the following linear time algorithm.

In order to recognize if a graph G is 3-colorable, we apply $T_G = dfs(G)$ and test if T_G has one of the following base cases:

1. If T_G has not odd fundamental cycles then G is 2-colorable. This option considers the case where T_G is a bipartite graph.
2. If any base cycle in T_G includes odd cycles, but such odd cycles are non-intersected with any other cycle of T_G then T_G is 3-colorable.
   We can color T_G by levels, but using the third color for coloring each end-node of every odd cycle. The end-node of a cycle is the node where the back edge was found during the depth-first search.
3. We color embedded cycles from the most intern to extern cycle. When we start to color a new cycle, we use a different color to their neighboring nodes (at most two neighboring nodes), since we consider that only the nodes in the internal cycles have already been colored. In this case, we can consider to T_G like an instance of a *Series-Parallel graph*, which is know that is colorable in polynomial time [9].

Then, if the topological structure of the input graph G is whatever of the latter basic cases, then G is 3-colorable and it is coloring in polynomial time. Thus, the class of graphs that they have the previous basic topologies, conforms us a new polynomial class of graphs for the 3-colorable problem.

We present in the following section a polynomial time algorithm based on the selection of the critical nodes to approximate the chromatic number.

4 A Polynomial Approximation Algorithm for the Chromatic Number of a Graph

First, we review some results that we will use to design our proposal, as well as the knowing upper bounds for the chromatic number of a graph.

Lemma 3 *If G is a connected non-regular graph, then $\chi(G) \leq \Delta(G)$.*
If G is regular, then $\chi(G) \leq \Delta(G) + 1$.

Proof. First, assume that G is non-regular, choose a vertex v_1 of minimum degree in V as the root of a spanning tree. We traverse the resulting graph in preorder (first the child nodes and after the parent node). When each vertex $v_i \neq v_1$ comes to be colored, the parent of v_i has not already been colored. Therefore at most $\delta(v_i) - 1$ adjacent vertices have already been colored. Hence $\chi(v_i) \leq \Delta(G)$. When v_1 comes to be colored, all adjacent vertices have already been colored. Since $\delta(v_1) < \Delta(G)$, we conclude that $\chi(v_1) \leq \Delta(G)$. Hence $\chi(G) \leq \Delta(G)$.

If G is a regular graph, then the proof proceeds as above, except that $\chi(v_1) \leq \Delta(G)+1$. The conclusion is followed. Notice that if G is regular, only one vertex needs to use color $\Delta(G) + 1$.

If a depth-first search is applied for ordering the nodes of a complete graph K_n, and after we color the nodes of the spanning tree in preorder, then it could be shown that it is needed to use exactly n colors.

Given a graph $G = (V, E)$ and a node $v \in V$, the subgraph induced by $N(V)$ is denoted as $H(v)$; recalls that it consists of $N(V)$, the neighborhood of v, and all edges in E between vertices of $N(v)$.

Lemma 4 *If G is k-colorable, then for any $v \in V$, $H(v)$ is $(k-1)$-colorable.*

Since 2-colorable graphs can be identified and colored (with only two colors) in polynomial time, the neighborhood of any vertex in a 3-colorable graph can be colored with two colors in polynomial time.

Let $G = (V, E)$ be a graph with $\chi(G) = m$. If we remove an edge $\{u, v\}$ from G, there are two possibilities, either $\chi(G-\{u,v\}) = m$ or $\chi(G-\{u,v\}) = m-1$. In the latter case, we say that the edge $\{u, v\}$ is critical. If a node u has a critical edge upon it, then we extend the definition and the node u is critical too.

A graph G is critical if $\chi(G - \{u, v\}) = \chi(G) - 1$ for all edges $\{u, v\} \in E$. If $\chi(G) = m$, we say that G is m-critical.

It is easy to see that every graph G contains a critical subgraph. If $\chi(G - \{u, v\}) = \chi(G)$ for some edge $\{u, v\} \in E$, we can remove such edge. Continuing deleting edges like this until every edge is critical. The result is a critical subgraph.

According to those latter results, we design the following algorithm.

Let $G = (V, E)$ be a graph with $|V| = n$, $|E| = m$, and let T_G be the graph generated for the depth-first search over G.

The general strategy of our proposal for coloring G consist on:
First: To recognize the general topology of G. This is done for applying a depth-first search on the graph.
Second: Test if G can be colored with two colors, a linear procedure is executed for detecting this condition
Third: If G is not 2-colorable, we detect the node v which is part of an odd cycle and with maximum conflicting for coloring v and the odd cycles in G. That is done by executed a polynomial time procedure.
Fourth: We color v with the active color, v and its edges upon it are deleted from the current graph. And the control is returned to the first step.

This procedure is executed in iterative way while the current graph can not be recognized as a bipartite graph. We show the pseudo-code of this proposal.

Procedure Select_Candidate_Node(T_G,NV)
Input: T_G is a subgraph, NV is the neighborhood over the vertices that can not be colorable with color k

Output: $v \in V$ **a vertex to be coloreable**
Procedure

1. $Vertices = minus(T_G, NV)$; /* Computes $Vertices = V(T_G) - NV$ */
2. choose $v \in Vertices$ such that /* v is a critical node */
3. degree(v) and degree($NV(v)$) are maximum over the set of odd cycles nodes
4. Otherwise /* If every odd cycle was covered by NV */
5. choose $v \in Vertices$ such that /* v has maximum degree in T_G */
6. If (maximumOver_$TG(degree(v), T_G)$ == true) then
7. Return v

Algorithm Seek_Chromatic_Number(G)
Input: G a non directed graph
Output: An approximate value for $\chi(G)$
Procedure

$k = 3$; /* Starting with the class color $k = 3$ */
$T_G = dfs(G)$; /* The nodes of the graph are ordered */
1) while(is_bipartite(T_G)==false) /* While there is an odd cycle in G */
{$NV = \emptyset$; /* NV is the neighborhood for the class color k */
2) while (is_subset(NV, $V(T_G)$)==true)
a) { $v = Select\_Candidate\_Node(T_G, NV)$;
b) $Color(v) = k$; $delete(T_G, v)$;
$add(NV, N(v))$; /* $NV = NV \cup N(v)$
c) H =Max_Component(T_G); /*If T_G is disconnected then consider the component with maximum value for $\chi(T_G)$*/
d) $T_G = dfs(H)$; /* Maintain ordered the remaining nodes */
} k++;
}
3) Call 2-Coloring(T_G); /*At the end, the remaining graph is 2-colorable*/
Return

The procedure *Seek_Chromatic_Number* consist of two embedded loops. In each iteration of the external loop, an independent set of class color k is built. This class color is formed by the critical nodes of the current graph and compose an independent set I_k of the graph. Each node in a independent set I_j is colored with the color $j + 2$, like it is showed in first and second graph in Figure 2.

The internal loop is applied to find each critical node v of the current graph and for building the neighborhood $NV(v)$ for the independent set I_j where v is, like it is showed in second graph in Figure 1. When a critical node is detected, it is colored and deleted from the graph as well as incident edges upon it; as T_G is changing in each iteration of the internal loop, it might even be disconnected, then it is necessary to order the nodes using the depth-first search again.

The external loop finishes when the remaining subgraph T_G is bipartite, and then T_G is 2-colorable, like it is showed in third graph in Figure 2.

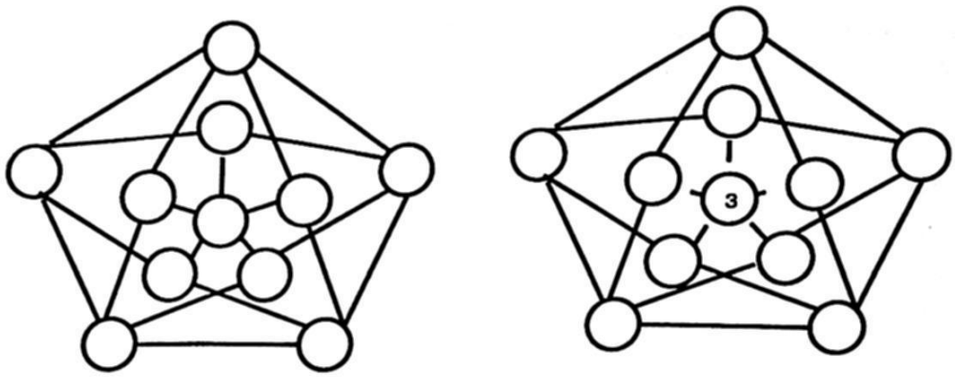


Fig. 1. Executing the algorithm over the Grötzsch graph

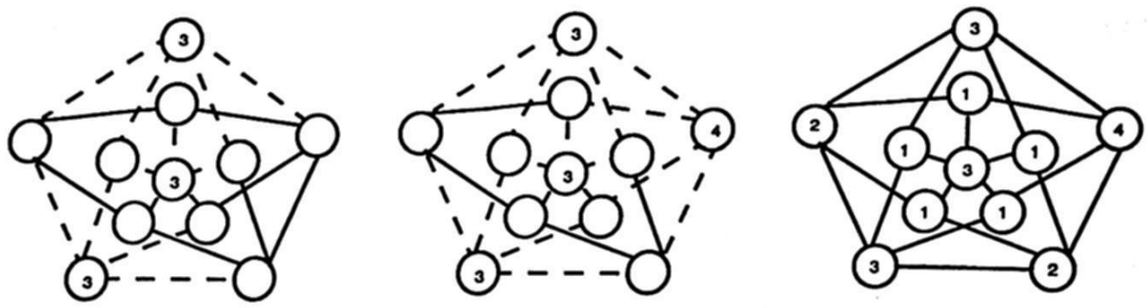


Fig. 2. Three iterations of the main loop

5 Complexity Analysis

The most expensive operation into the internal loop (step 2) is to determine the critical node v to be colored, line (a). This step is performed in time $O(m * n)$, and as that loop iterates at most n times, then the complexity time of the internal loop is $O(m * n^2)$. While the external loop (step 1) is executed $(m - n)$ times in the worst case (when all cycle in G is odd), and the most expensive instruction is the internal loop, so the total complexity time of the procedure is polynomial and it is $O((m - n) * m * n^2)$.

We focus ourselves on determining the average number of colors used by our algorithm, rather than the worst-case behavior. Let $rel = m - n$ be the variable used to denote the relation between the number of edges and nodes in whatever subgraph. The variable rel denotes the number of cycles minus 1 in a connected graph. We know that for any graph H, if $m \leq n$ then H is bipartite or H is a simple odd cycle and then H is 3-colorable. We analyze the value that rel_i takes in each subgraph $T_{G_i} \subset G$ generated after of the iteration i by the main loop (external loop), and let K_i be the independent set built in such iteration.

Given an initial graph $G = (V, E)$ with $|V| = n_0$ and $|E| = m_0$ in each iteration of the main loop the number of nodes and edges are updated as: $n_{i+1} = n_i - |K_i|$ and $m_{i+1} = m_i - |E(K_i)|$, since in each iteration the nodes in K_i are deleted to the current graph as well as its incident edges: $E(K_i)$.

Let $G_{i+1} = G_i - K_i$ be the remaining subgraph generated from G_i after to finish the iteration i into the main loop. As each K_i is an independent set of G_i, there are not edges of G_i connecting any two nodes of K_i. Furthermore, the edges in $E(K_i)$ cover every node of G_i, that is, $E(K_i)$ is an edge cover of G_i, then we have that $|E(K_i)| \geq n_i$ and the number of remaining edges in G_{i+1} is $m_{i+1} = m_i - |E(K_i)| \geq m_i - n_i$, so $m_i - n_i$ is an upper bound for m_{i+1}.

The behavior between the number of edges and number of nodes for each subgraph G_i, hands:

$rel_0 = m_0 - n_0$.
$rel_1 = m_1 - n_1 \leq (m_0 - n_0) - n_1 = (m_0 - n_0) - (n_0 - |K_0|) = m_0 - 2n_0 + |K_0|$.
$rel_2 = m_2 - n_2 \leq (m_1 - n_1) - (n_1 - |K_1|) = m_0 - 2n_0 + |K_0| - (n_0 - |K_0| - |K_1|) = m_0 - 3n_0 + 2|K_0| + |K_1|$.
$rel_3 = m_3 - n_3 \leq (m_2 - n_2) - (n_2 - |K_2|) = m_0 - 3n_0 + 2|K_0| + |K_1| - (n_0 - |K_0| - |K_1| - |K_2|) = m_0 - 4n_0 + 3|K_0| + 2|K_1| + |K_2|$.
...
$rel_k \leq m_0 - (k+1)n_0 + k|K_0| + (k-1)|K_1| + \ldots + |K_{k-1}|$.

An average value for $|K_i|, i = 0, \ldots, k-1$ is computed for considering that $\sum_{v \in K_i} \delta(v) \approx \sum_{j=1}^{|K_i|} \delta(G_i) \geq n_i$, where $\delta(G_i)$ is the average degree of the subgraph G_i, and then $|K_i| \cdot \delta(G_i) \geq n_i$ so $|K_i| \geq n_i/\delta(G_i) \approx n/\delta$, being δ the average degree of the initial graph G. Thus, we can approximate the value $|K_i|$ by n/δ.

Then, $rel_k \leq m_0 - (k+1) \cdot n_0 + \sum_{i=0}^{k-1}(k-i) \cdot (n/\delta) = m_0 - (k+1)n + (n/\delta) \cdot ((k)(k+1)/2)$. And we want to know the average number of iterations for the external loop until arrive that $rel_k \leq 0$. So, we want to determine the average value for k where $rel_k \leq 0$, that is, $m_0 + (k(k+1)/2) \cdot (n/\delta) \leq (k+1)n_0$.

Let $m = m_0$ and $n = n_0$. As $m = (\delta \cdot n)/2$ then

$[(\delta \cdot n)/2 + (n/\delta) \cdot (k(k+1))/2]/n \leq k+1$, so $\delta + (k(k+1))/\delta \leq 2 \cdot (k+1)$. Thus,

$$\delta/(k+1) + k/\delta \leq 2 \tag{1}$$

In order to solve (1) we express the equation in terms to the variable k, obtaining: $0 \leq -(1/\delta)k^2 + (2 - (1/\delta))k + (2 - \delta)$ and factoring the polynomial in k, we have: $0 \leq -(1/\delta)(k + (\sqrt{4\delta+1} + 1 - 2\delta)/2)(k - (\sqrt{4\delta+1} + 2\delta - 1)/2)$, and the interval where the value for k hands the equation (1), is: $-(\sqrt{4\delta+1} + 1 - 2\delta)/2 < k < (\sqrt{4\delta+1} - 1 + 2\delta)/2)$. Thus, the maximum value for k handing (1) and which represents the average number of colors used for our algorithm, is: $\chi(G) \approx (\sqrt{4\delta+1} + 2\delta - 1)/2$.

Hence our procedure uses $\tilde{O}(\delta)$ colors (where the notation $\tilde{O}$ is used to suppress polylogarithmic factors) for coloring the input graph G, being δ the initial average degree of the graph G.

6 Conclusions

First, we show a new polynomial class of graphs for the 3-coloring problem which includes to the *Series-Parallel graphs* like instances of this class. After, we show a new approximation polynomial-time algorithm for determining the chromatic number of a graph G. Our proposal is based on selecting, in iterative manner, a critical vertex v of the graph. The criterion to select is based on choose the node with maximum degree and with maximum degree of its neighborhood into the set of vertices composing odd cycles.

Our algorithm establishes an average number of $(\sqrt{4\delta+1}+2\delta-1)/2$ colors for approximate the chromatic number $\chi(G)$ for any input graph G, being δ the average degree of the graph G. Hence our procedure uses $\widetilde{O}(\delta)$ colors for coloring an input graph G.

References

1. S. Arora, E. Chlamtac, M. Charikar, New Approximation Guarantee for Chromatic Number, *Proceedings STOC 2006*, May 2006.
2. Baase S., Gelder A. V., *Computer algorithms: Introduction to Design & Analysis*, Addison Wesley, 2000.
3. Blum A., Karger D., An $O(n^{3/14})$-coloring algorithm for 3-colorable graphs, *Information Processing Letters*, 61(1):49-53, 1997
4. Bellare M., Sudan M., Improved Non-approximability results, Draft, 1993.
5. Dyer M., Greenhill C., *Some #P-completeness Proofs for Coulorings and Independent Sets*, Research Report Series, University of Leeds, 1997.
6. Dyer M., Greenhill C., Corrigendum: The complexity of counting graph homomorphism, *RSA: Random Structures and Algortihms*, 25:346-352,2004.
7. Greenhill Catherine , The complexity of counting colourings and independent sets in sparse graphs and hypergraphs, *Computational Complexity*, 1999.
8. Kocay W., Kreher D., *Graphs, Algorithms, and Optmization*, Chapman & Hall/CRC Press, 2005.
9. Johnson D., The NP-Completeness Column: An Ongoing Guide, *Jour. of Algorithms* 6,434-451, 1985.
10. Roth D., On the hardness of approximate reasoning, *Artificial Intelligence 82*, (1996), 273-302.
11. Russ B., *Randomized Algorithms: Approximation, Generation, and Counting*, Distingished dissertations Springer, 2001.
12. Vadhan Salil P., The complexity of Counting in Sparse, Regular, and Planar Graphs, *SIAM Journal on Computing*, Vol. 31, No.2, (2001), 398-427.
13. Wigderson A. Improving the performance guarantee of approximate graph coloring, *Jour. of the ACM*,30 (4):729-735, 1983.

Natural Language Processing

Semantically Mapping the Web

Eduardo Ramirez and Ramon Brena

Tecnologico de Monterrey, Mexico

Abstract. The millions of web pages populating the internet seem to be unstructured and chaotic, but there are implicit semantic relations between them. In this paper we propose to make explicit the underlying semantic structure of the internet, by measuring joint keyword occurrences in web pages, around our notion of "Semantic Contexts". As a result, we can draw a "map" of semantic clusters which can be used as a reference for situating individual web pages in a complex semantic space. Further, the methods we propose could be used for disambiguating and refining web search queries, for refining translations, for spam filtering, and in general for semantic-enabling many internet applications.

1 Introduction

Internet is acknowledged as one of the big technological revolutions of our time; since its inception in the early 90s, the WWW has grown exponentially, reaching some 74.5 millions of websites with at least 11.5 billions indexed at the main web searchers [1]. Nevertheless, web pages normally have the limitation of not taking into account the meaning or the context of the included information content, but just its formatting. HTML tags indicate that a certain text is a title, or a series of items, but not what the document is about. In words of T. Berners-Lee, "Most of the Webs content today is designed for humans to read, not for computer programs to manipulate meaningfully" [2]. This is indeed a serious limitation; for instance, one very important issue is to determine what a given web page is about. The lack of an efficient semantic categorization undermines many internet applications, in particular web searches. Indeed, every internet user is confronted with the inconvenience of receiving from the search engines many irrelevant pages, due to the inability of search engines to contextualize keywords in meaningful concepts, areas, themes, etc.

Initiatives aiming to represent in web pages meaning, have been generically called "Semantic Web" [2] The Semantic Web initiative proposes markup languages, mainly based on XML [3], and develops technologies for defining and using concepts and relations among them in the so-called "ontologies".

Nevertheless, there has been problems for widely adopting semantic web. Some of the reasons are technical challenges, and other are practical issues. So we are turning our attention to quantitative approximate methods (sometimes called "soft" [4]) for characterizing internet semantic relations. In particular, we proposed to exploit the joint frequencies of keywords as representative of semantic closeness in the existing internet, not in an ideal or futuristic internet.

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 125-136

Received 24/02/07
Accepted 08/04/07
Final version 17/04/07

We propose in this paper a keyword-based quantitative semantic infrastructure that could make explicit an underlying internet semantic structure, as a collection of interrelated "Semantic Contexts", which constitute a sort of internet semantic "topography". The Semantic Contexts are a stable reference against which specific web pages or queries could be situated. In this paper we also present some practical applications of Semantic Contexts, in particular how to better focus web searches.

After this introduction, in the next section we give a technical presentation of our method, followed by some experimental results, then by a representative application, and then a comparison with related work, to end with a conclusion.

2 Our proposal

The basis of our approach is to make a semantic interpretation of joint keyword frequency. Central to our approach is the notion of "Semantic Contexts" (SC), which intuitively represent conceptual areas, around which a family of keywords appears frequently inside web pages. For instance, around a concept of "Tourism" there will be many keywords like hotels, reservations, flights, etc., which appear together in many web pages.

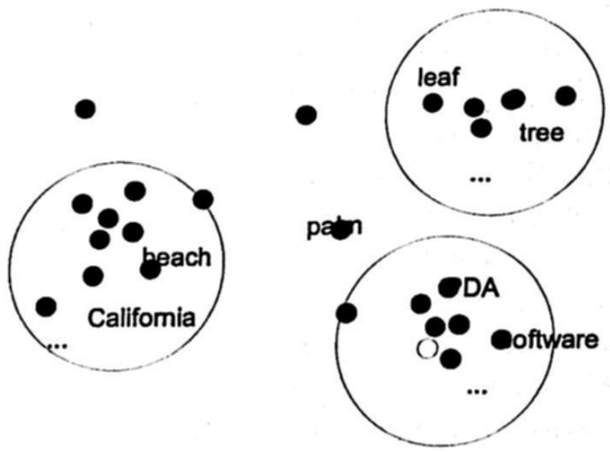


Fig. 1. Semantic Contexts related to "palm"

SC are defined formally in the next section, but let us first introduce a motivating example. Imagine the following scenario: a user is trying to find information about how to display pictures on a handheld device, so he/she issues the search query palm pictures. In a standard search service, like Google [5], this query would throw to the user answers about topics like: display of pictures on a handheld device, pictures of a place in California, pictures of an unbranched evergreen tree, etc. In a semantically-enhanced version of web search, the system would consult a base of indexed semantic clusters and would offer the user the following options: Search for palm pictures related to: 1) pda, software, 2)

California, beach, or 3) tree, leaf. Once the users would select one of these options, only pages of the corresponding interpretation of the word palm would be returned to the user. This, of course, would be of enormous utility to users, because search results would be much more focused.

2.1 Semantic Contexts

We see SCs as sets of interrelated keywords that appear together in a number of pages. We can visualize SC intuitively as "clouds" in a space of semantic closeness. like depicted in figure 1, for the "palm" example we just presented.

In order to formalize the notion of SCs, we consider the relative weights w_i of keywords k_i as a measure of how important these keywords are in a given topic. For instance, when we are talking about coffee, other related words like "sugar" or "roasted" have a high weight. We assume weights are normalized to be in the range $0 \leq w_i \leq 1$. Given a set K of n keywords, we define a SC as a function $\sigma : K \rightarrow [0, 1]$. As a SC represents a "topic", "subject" or "theme", important words in that theme have higher weights. We could also imagine SC as vectors $w_1, w_2, \ldots, w_n$ of weights for keywords $k_1, k_2, \ldots, k_n$.

Distances between SC can be readily calculated. First, we define SC similarity using a standard internal product formula [6]. Assume the vector $w_1, w_2, ..., w_n$ of weights for keywords $k_1, k_2, ..., k_n$ in a given semantic context SC is written as $\mathbf{w}$. Then, we can take the internal product of vectors as a similarity measure:

$$sim(SC_1, SC_2) = \frac{\mathbf{w_1} \bullet \mathbf{w_2}}{|\mathbf{w_1}|\,|\mathbf{w_2}|} \tag{1}$$

Then, from a similarity measure we could take the cosine inverse to obtain a difference measure as an angle [6]. Other similar distance metrics could be readily defined.

Next is the question of how a SC can be calculated. This can be done using specialized conjunctive queries, which we call "*k-cores*".

k-cores Now we introduce the notion of "k-cores", which are conjunctive queries, as follows. $F(w)$, called *frequency*, will represent in how many corpus pages the term w apears. By extension, the notation $F(\{w_1, \ldots, w_k\})$, for sets of k keywords $\{w_1, \ldots, w_k\}$, represents the count of web pages where all of $\{w_1, \ldots, w_k\}$ appear together (in the same document). Further, we use $F(\mathcal{P}\{w_1, \ldots w_k\})$ which is the set of frequencies, one for each subset of $(\mathcal{P}\{w_1, \ldots w_k\}$.

Then we define the "force" f of a keyword set $\{w_1, \ldots, w_k\}$ as follows, where c is a suitable constant, like 10^{12}; function g is explained below:

$$f(\{w_1, \ldots, w_k\}) = c\frac{F(\{w_1, \ldots, w_k\})}{g(F(\mathcal{P}(\{w_1, \ldots, w_k\})))} \tag{2}$$

and g is a function of joint frequencies of subsets of $\{w_1, \ldots, w_k\}$. One such function is the "disjoint frequency", which is the quantity of pages where a

given set of keywords (in this case $\{w_1, \ldots, w_k\}$) does *not* appear together, but some of $w_1, \ldots, w_k$ does appear.

Now we define "k-cores" as sets of k keywords of maximal force, meaning that replacing just one of its keywords by any other available word will decrease the force. k-cores can be seen as local maxima in a space of sets of keywords. This naturally suggests hill-climbing [7] as a method for finding them.

Depending on the application, the size k of k-cores could take different values. Of course, any value smaller than 2 does not make any sense, and even a value of 2 will normally be too small to represent a meaningful theme. In our experiments we use mostly a value of $k = 4$. Choosing the "right" value of k is an open question right now, and we have been rather pragmatic on this issue, generally taking a value of 4, with which the experiments gave meaningful results (see future work at the end of this paper).

k-cores are a key component of our method. They are conjunctive queries that represent a topic or subject. Once a k-core ω is determined, given a certain corpus C, the subset Ω of C with documents containing simultaneously all of the keywords in ω, can be readily obtained using web indexing technology [6]. From Ω, keyword weights can be computed using standard tf-idf measures [6], with a formula like:

$$w_{x,j} = f_{x,j} \times \frac{idf_x}{max_i\ idf_i} \tag{3}$$

where $f_{x,j}$ is the normalized frequency of term k_x in document d_j, and idf_i is the inverse document frequency for a generic term k_i.

In order to "mine" a set of web pages for finding k-cores, there is a trivial hill-climbing algorithm, as follows:

1: *Input*: A set P of web pages and a number k (for calculating size-k k-cores).
2: *output*: A set S_k of k-cores.
3: From P filter a set W of keywords.
4: $S_k \leftarrow \emptyset$ –The set of k-cores is initially empty.
5: **repeat**
6: $K \leftarrow$ a random subset of W of size k.
7: $F \leftarrow f(K)$
8: **for all** $w \in W - K$ **do**
9: **for all** $wk \in K$ **do**
10: $K' \leftarrow$ replace w for wk in K
11: $F' \leftarrow f(K')$
12: **if** $F' > F$ **then**
13: $F \leftarrow F'$; $K \leftarrow K'$
14: **end if**
15: **end for**
16: **end for**
17: $S_k \leftarrow S_k \cup K$
18: **until** k-cores are "stable"

The condition at the end of the **repeat** loop means that there are no changes in the set of current k-cores, meaning that this set is a fixpoint of the algorithm.

In practice, for efficiency reasons, this condition could be replaced (and actually has been replaced in our experiments) by a fixed number of iterations.

In order to assess the complexity of an algorithm we can evaluate first the inner loops. From the first *for* loop, the call to f function in the inner loop (which is normally the most costly operation) will be executed $|W - K|k$ times, and considering that k is kept constant and that $|W - K|$ is basically $|W|$, we can see that this algorithm is linear in the keyword set size $|W|$. This result stands of course if the outer loop is replaced by a fixed number of iterations, like we do in the experiments presented below.

We have introduced a small optimization to this algorithm: instead of starting with random cores, we select "promising" cores obtained in the following way:

1: From a random page p in the corpus we obtain the k most relevant terms using a TF-IDF measure [6, 8]
2: The starting k-core is the set of those k terms.

Mining a corpus for k-cores can be seen as locating the "topics" to which documents belong at least partially. We view the set of k-cores as the summits in a semantic topography, where altitude is calculated by the "force", given by equation 2. k-core calculation could be a computationally costly process, but it would be done offline in servers, so it does not affect the performance with respect to user queries, which we present in the following section. In the experiments section we show an example of k-cores calculation in a controlled environment.

3 Experiments

In order to validate the ideas presented above, we setup an experimental framework described in the following.

We installed an indexer and web searcher (Apache Lucene, [9]), and gather a small collection of 1168 web pages in the following topics: investments, java development, architecture, music, middle ages history and travel and tourism.

In order to provide an objective basis for classifying pages in topics, we used the Google and Yahoo directories, and using the APIs of these services for automatic downloads, avoiding in this way to introduce an involuntary bias. Of course, the Google and Yahoo directories were made by humans as well, but at least they were made by many people, and not including ourselves.

Then we ran the indexer in order to enable web searching inside our controlled set of web pages. The indexer created the index file and a table of keyword frequencies. We had a set of 50,025 words. In order to consider only meaningful keywords, we performed an automated filtering of "stop-words" (meaningless words, like "above", "etc", etc). 12,745 terms were filtered out, which is about a quarter of the total, leaving 37,280 keywords.

The next step was removing variants of the same words, like run, running, etc; this process is known as "stemming" [10]. In our prototype we used a stemming algorithm provided by the "Snowball" implementation of [11], which is not part of this research. We also added a database of similar words like "built" and "build" that were not caught by the Snowball system, so when the replacement

algorithm find a word similar to one in the current core, only is replaced if it is the word we are currently replacing, and the force is increased, otherwise it is discarded and we continue with the algorithm as presented in section 2.1.

Even in a small document sample like the one we have, with just over 1000 pages, the quantity of possible cores of a size 4 or 5 is quite impressive: there are 80,467,864,076,000,270 combinations of 37,280 words taken in groups of 4. This is the number of possible 4-cores, which of course excludes any brute-force algorithm for finding the best cores.

Actually, most of the cores have a force of exactly 0, because the numerator of the force formula 2 is the number of pages simultaneously having all of the considered keywords in it. The space of all possible cores contains a few (comparatively) sparse non-zero cores. In previous papers [12] we have found that the proportion of non-zero-force cores is about 0.018 percent. Taking into account this huge proportion of zero-force cores we can see that any refinement which avoid considering 0-force candidates would be a great improvement. We are using TF-IDF measures [8] of keyword relevance for selecting the best candidates, as we pointed out before. Consider that any word participating in a 4-core would necessarily be in some 2-cores (that is, sets of 2 words). For instance, two words appearing each in just 10 pages have a probability of appearing together in a given page of 7×10^{-8}, so it could be discarded. In our implementation we are forming initial cores ("seeds") by first selecting randomly one page in the corpus, and then selecting the 4 highest-valued words taking an TF-IDF measure. For instance, the most relevant words of a randomly selected web page, which was about japanese architecture, were *shinden*, *domestic*, *zukuri*, and *architecture*. We take this as a "seed" for the hill-climbing algorithm.

For the experiments of this paper, we implemented a variation of the algorithm in section 2.1, implemented in a "horizontal" way, meaning that we first calculated a single round of force increment, from initial seeds, and then calculated the second round from current cores, and so on. instead of going all the way to the maximum from initial seeds. This experiment is exhaustive for the small corpus we took, because initial seeds were calculated for every single page in the collection of over 1000 pages. In figure 2 we present the way the force of cores gets incremented from 2 rounds up to 6 rounds. We can see there that from 4 rounds-on variation is minimal, meaning that in practice there is no point repeating the "repeat" loop of algorithm in section 2.1 more than 4 times.

To illustrate the process, in the following table we present results from the "horizontal" hill-climbing algorithm, showing how many pages in the corpus, which produced corresponding initial seeds, are "concentrated" in the same 4-core. This means that some cores get frequently "merged" into the same core, resulting in gradually fewer and fewer cores as the algorithm proceeds.

As we can see in this table, for example the seed "architecture, building, design, house", which initially had just one page, then received the contribution of other cores that ended becoming identical to it by word substitution (see the algorithm), and had 52, 62 and finally 64 pages represented by it.

Core	*rnd 2*	*rnd 3*	*rnd 4*	*rnd 5*	*rnd 6*	Size var
{architecture,building,design,house}	1	52	62	64	64	63
{hotels,paris,rooms,rue}	57	83	91	92	92	35
{ages,life,middle,people}	59	88	90	90	90	31
{buffett,chairman,letter,warren}	6	25	25	25	25	19
{calderon,omar,reggaeton,tego}	17	22	22	22	22	5
{ages,feudal,middle,religion}	28	31	31	31	31	3
{band,blues,dance,jazz}	15	18	19	19	19	4
{application,code,developers,java}	5	9	12	12	12	7
{investors,premium,shares,stock}	0	6	9	9	9	9
{attractions,hotels,tourism,travel}	3	8	9	9	9	6
{berkshire,buffett,chairman,warren}	2	8	8	8	8	6
{cheap,hotel,reviews,star}	10	12	12	12	12	2
{artist,blues,jazz,pop}	1	5	5	5	5	4
{maze,mazes,pyramids,sphinx}	2	0	0	0	0	-2
{architectural,design,house,style}	2	0	0	0	0	-2
{ages,medieval,middle,weapons}	2	0	0	0	0	-2
{building,gate,great,middle}	2	0	0	0	0	-2
...						...

In the lower part of the table (below the horizontal line) we have some examples of cores which lost pages in the process, giving them away to stronger cores. These can be considered as meaningless word combinations in the corpus. The dotted last row represents the remaining 28 initial seeds of the experiment. Above the horizontal line we have all the 13 cores that ended with a positive size variation, which could be considered as the possible topics of the corpus.

Table 1. Cores found in our corpus

Topic	Cores
Investments	buffett,chairman,letter,warren
	investors,premium,shares,stock
	berkshire,buffett,chairman,warren
Programming	application,code,developers,java
Travel	hotels,paris,rooms,rue
	attractions,hotels,tourism,travel
	cheap,hotel,reviews,star
Music	calderon,omar,reggaeton,tego
	band,blues,dance,jazz
	artist,blues,jazz,pop
Architecture	architecture,building,design,house
Middle-ages history	ages,life,middle,people
	ages,feudal,middle,religion

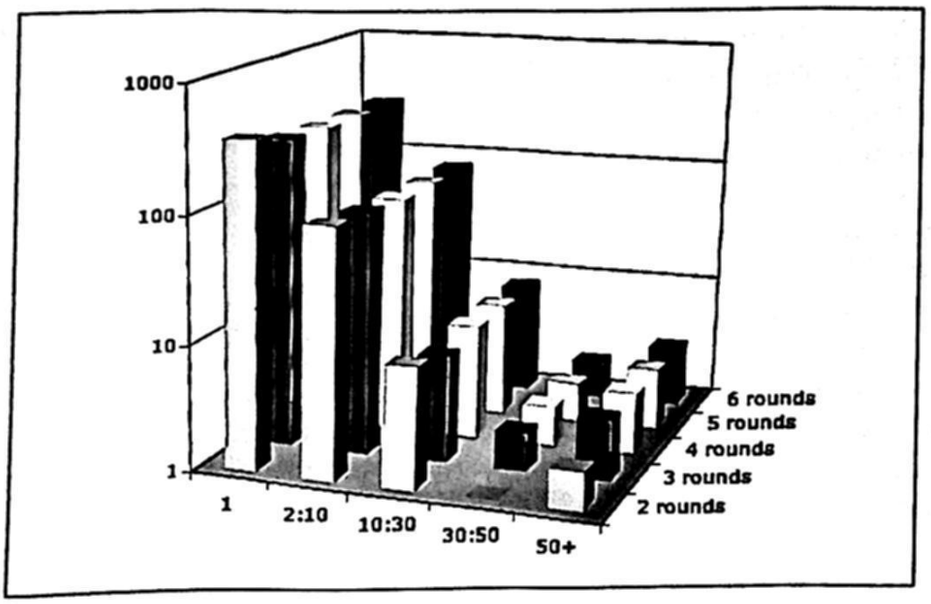


Fig. 2. Evolution of best core force through several rounds

Because we know in advance the topic of every page in the corpus, we can manually classify the 4-cores with positive variation into the 5 topics of the corpus, to see how the former represent the latter. This is done in table 3. From this table we can see that *all of the 5 topics got represented by at least one 4-core.* This is an important experimental result, because with it we showed that *the method was able to find all the relevant topics of the corpus.*

Nevertheless, some of the topics were represented by more than one 4-core. We believe that some of the cores are actually representative of a subtopic, like for instance "calderon, omar, reggaeton, tego", which is a subtopic of music. In order to test this hypothesis, we calculated the distances between all of the 13 SCs found, to see whether distances between semantically-related SCs were indeed smaller than distances between unrelated SCs. This is done in figure 3.

For this experiment we executed each of the cores as queries to the index, and then generated a weight vector using the resulting documents for each of the 13 selected cores, The weight of each term is its relative frequency in the result set. Then, the weight vector is unit-normalized so that vector distance metrics can be applied to each of the vectors in the set. As we can see in figure 2, k-cores in the same topic, like "buffett, chairman, letter, warren", "investors, premium, shares, stock" and "berkshire, buffett, chairman, warren" (all about the "investments" topic) are strikingly close to each other. The same could be said about same-topic k-cores in the other topics. Take, for instance the music-related k-cores: "band, blues, dance, jazz" is very close to "artist, blues, jazz,pop", and even "calderon, omar, reggaeton, tego", which does not share a single keyword with the other two k-cores, appears as semantically close in the figure. This validates the hypothesis that k-cores refer to subtopics of a general theme.

4 Application of SC to internet search

Now let us assume a set of web pages has already been mined for its k-cores, which will be considered each a representative of a SC. We will show how this structure could be used in order to guide an internet search.

The relevance of a keyword set or query Q to a given SC with k-core K and weights w_k, written as $R(SC_i, Q)$ is defined as the average weight w_k of the words $k \in Q$ in the SC_i.

The next step in SC-guided search is to calculate the relevance of Q with respect to the available SCs $SC_1 \ldots SC_n$. We order the SC Σ_i in descending order of relevance, and we take the first m relevant SC, where m is a small number like 2 or 3. These first m SC will be considered as the closest to the user query, and the associated k-cores will be presented to the user to choose from like in the introductory example.

Once the user selects one of the proposed k-cores, say K_i, the system will propose to him the result of queries $Q \cup \{k_i, k_j, \ldots\}$, where $k_i, k_j, \ldots$ are members of the selected k-core. This means that user queries can be enriched with words from the core, so that the search is narrowed. Notice that this will have a more restricted result than the original search Q, which is the intended effect.

Resuming, the algorithm for a SC-guided search is as follows (we assume that all k-cores satisfying a force threshold have already been calculated):

1: *Input*: A set SC_i of SC and a query Q.
2: *output*: Results from an enriched query.
3: Calculate the relevances $R(SC_i, Q)$ to SC, which are Σ_i.
4: Construct a list L of SC with decreasing relevance to the query.
5: Present to the user the m first k-cores from the SC in L.
6: The user selects one of the k-cores of the preceding step, let it be K_j.
7: New queries of the form $Q \cup \{k_i, k_j, \ldots\}$, where $k_i, k_j, \ldots$ are members of K_j, are constructed.
8: The user receives the result of the enriched queries.

Our experiments about SC-enhanced search are reported elsewhere [12].

5 Related work

As we mention in the introduction, there are a number of quantitative corpus-based approaches to analyze texts [13, 14], but none of them offers, as we do, a perspective of semantically structuring the web space; the cited works belong to the Natural Language Processing field.

In the field of Information Retrieval some works propose to extend search engines functionalities [15], by different means than the ones proposed in our paper, like "retrieving any type of data and collecting information to do better web mining", and other improvements like dealing with multimedia data. The cited author does mention the use of "Soft Computing" methods [16], but without proposing a specific approach or application.

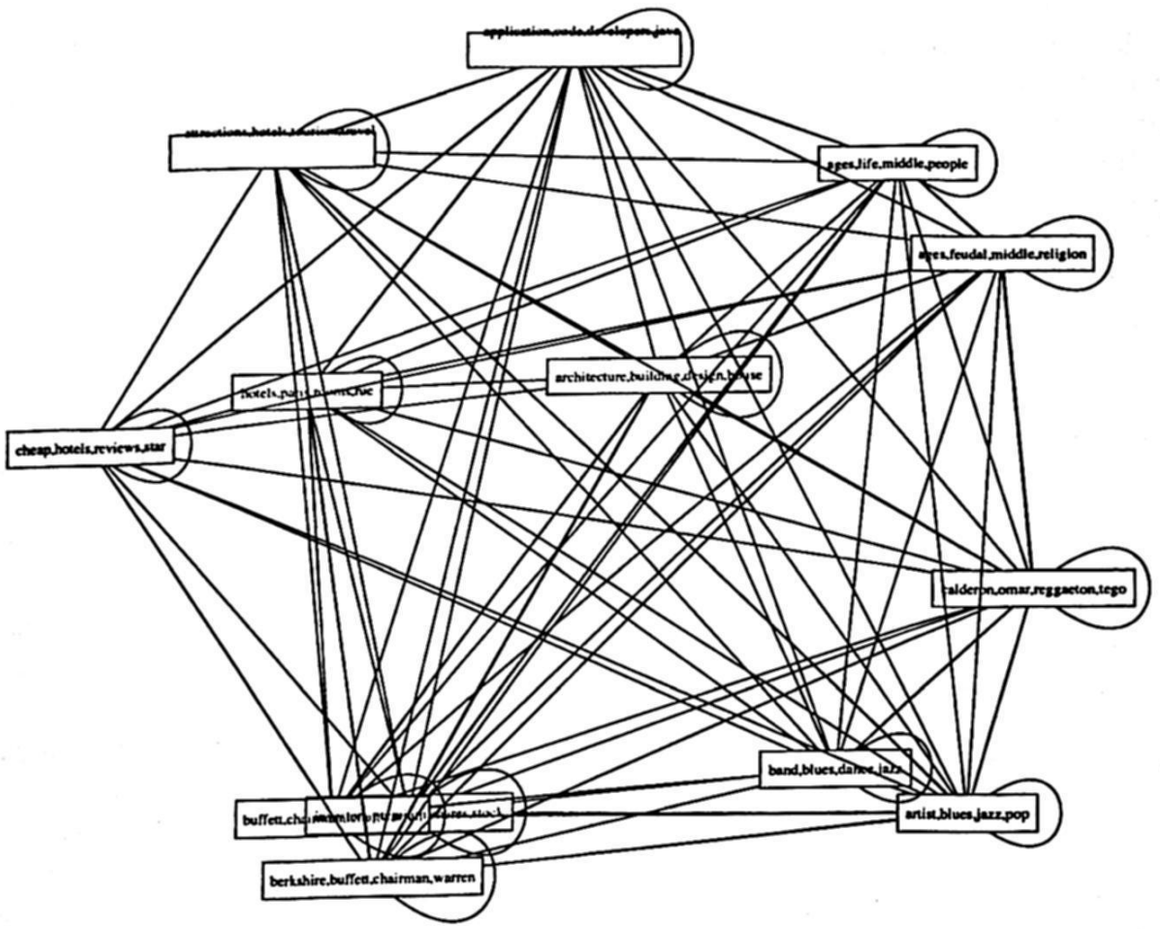


Fig. 3. Distances between SCs

In [17] quantitative information-theoretic measures of Semantic Similarity are explored using a tree-based notion of semantic similarity. Our work does not rely on graph comparison, but entirely on joint frequency measures, which are efficiently calculated by web search engines.

Some other works [18–20] propose clustering methods for sets of documents. For instance, in [21] "chat" sessions are put in relation to possible contexts using the web as a reference corpus; the author uses a clustering algorithm to identify candidate contexts. In his approach, a web search is done first, and clustering is applied to the search results. Our approach is not to directly cluster document sets, or search results like [21] or the Clusty search and clustering engine [22], but instead to first mine a corpus, that could be a document set or the whole internet, for Semantic Contexts, represented by their k-cores, and only then, match documents or queries against the k-cores; this last step is done efficiently using algorithms very similar to those used by search engines, which

were described in section 4. One advantage of doing so is that our k-cores are static, that is, they do not change from query to query, but only through years of internet evolution, and thus they can be calculated off-line, that is, prior to user querying, reducing this way the user waiting time.

A work in NLP similar in ideas to our work is [23], where the author presents a vector representation of keyword occurrences together. Topics are represented by the centroid of a set of vectors in a multidimensional space. There are complexity issues though, as the author declares: "a global optimization of cooccurrence constraints is necessary, an operation so complex that only a supercomputer can perform it". Our reliance on web search technology, in contrast, gives us, we think, better chances to scale up to the whole internet.

6 Conclusion

As we show in this paper, part of the underlying semantic structure of the web could be made explicit by means of our "Semantic Contexts", represented each by corresponding keyword weights and "k-cores". We have presented the notion of Semantic Context as "clouds" in a keyword space, we have formally defined them as weighting functions over keywords, and we have shown how they can be calculated. Further, the experiments we present show that it is possible to produce automatically k-cores representing all of the topics in the given corpus. As we are able to define distances over SC, we see the collection of SC in a corpus like a "map" of its semantic concentration points, or as a semantic "topography", with summits associated to cores with maximal force.

As a practical application of Semantic Contexts, search engines utility could be improved, using semantic contexts as a guidance. We think SC could be applied to automatic "tagging", to natural language translation, and in general to serve as an objective semantic reference that could make semantic-aware many internet applications.

To the best of our knowledge, our work, proposing the explicit construction of a static interrelated collection of semantic themes representative structures (Semantic Contexts and their k-cores), and their application for refining searches, is completely original.

Our future work includes a larger scale validation, the refinement of the algorithms to ensure scalability, the investigation of the effect of adjusting the k size of k-cores, as well as developing practical applications like the search refinement.

Acknowledgement: This work was supported by the CAT-011 Monterrey Tech's research chair.

References

1. Gulli, A., Signorini, A.: The indexable web is more than 11.5 billion pages. In Ellis, A., Hagino, T., eds.: Proceedings of the 14th international conference on World Wide Web, WWW 2005, Chiba, Japan, May 10-14, 2005 - Special interest tracks and posters, ACM (2005) 902–903

2. Berners-Lee, T., Hendler, J., Lassila, O.: The semantic web. Scientific American (2001)
3. Andersen, A.: Construction of XML. XML journal (2001)
4. Zadeh, L.: Soft computing and fuzzy logic. Software, IEEE **11** (1994) 48–56
5. google. (`http://www.google.com`)
6. Baeza-Yates, R., Ribeiro-Neto, B., et al.: Modern information retrieval. Addison-Wesley Harlow, England (1999)
7. Russell, S., Norvig, P.: Artificial Intelligence a Modern Approach. AI. Prentice_Hall (1995)
8. Church, K., Gale, W.: Inverse document frequency (IDF): A measure of deviations from Poisson. Proceedings of the Third Workshop on Very Large Corpora (1995) 121–130
9. lucene. (`http://lucene.apache.org/java/docs/`)
10. Xu, J., Croft, W.: Corpus-based stemming using cooccurrence of word variants. ACM Transactions on Information Systems (TOIS) **16** (1998) 61–81
11. Porter, M.: An algorithm for suffix stripping. Program **14** (1980) 130–137
12. : (Self-citations excluded for blind review)
13. Brill, E., Mooney, R.J.: An overview of empirical natural language processing. The AI Magazine **18** (1998) 13–24
14. Ng, H.T., Zelle, J.M.: Corpus-based approaches to semantic interpretation in NLP. AI Magazine **18** (1997) 45–64
15. Baeza-Yates: Information retrieval in the web: Beyond current search engines. IJAR: International Journal of Approximate Reasoning **34** (2003)
16. Zadeh, L.A.: Fuzzy logic, neural networks, and soft computing. Communications of the ACM **37** (1994) 77–84
17. Maguitman, A., Menczer, F., Erdinc, F., Roinestad, H., Vespignani, A.: Algorithmic computation and approximation of semantic similarity. WWW Journal (2006)
18. Zhao, Y., Karypis, G.: Evaluation of hierarchical clustering algorithms for document datasets. In Kalpakis, K., Goharian, N., Grossmann, D., eds.: Proceedings of the Eleventh International Conference on Information and Knowledge Management (CIKM-02), New York, ACM Press (2002) 515–524
19. Steinbach, M., Karypis, G., Kumar, V.: A comparison of document clustering techniques (2000)
20. Adami, G., Avesani, P., Sona, D.: Clustering documents in a web directory. In Chiang, R.H.L., Laender, A.H.F., Lim, E.P., eds.: Fifth ACM CIKM International Workshop on Web Information and Data Management (WIDM 2003), New Orleans, Louisiana, USA, November 7-8, 2003, ACM (2003) 66–73
21. Segev, A.: Identifying the multiple contexts of a situation. In: Proceedings of IJCAI-Workshop Modeling and Retrieval of Context (MRC2005). (2005)
22. clusty. (`http://www.clusty.com`)
23. Schutze, H.: Dimensions of meaning. In: Proceedings Supercomputing'92, Minn., MN, IEEE (1992) 787–796

Keywords Extraction in Clusters of Related Documents

Leticia Arco, Damny Magdaleno, Rafael Bello,
Manuel Llanes and Libernys Valdés

Central University of Las Villas, Carretera a Camajuaní km 5 ½,
54830 Santa Clara, Villa Clara, Cuba
{leticiaa, dmg, rbellop, manuela, libernys}@uclv.edu.cu

Abstract. The aim of this work is to develop a model that allows the application of feature selection techniques for the extraction of relevant terms that characterize the clusters of related documents and discriminate among clusters. The main feature selection techniques are described as well as their applications to text mining, particularly the induction of decision trees in feature selection. We outline a flexible model that justifies the design and subsequent application of the stages that make up the proposed procedure, which are: the discretization of the features that describe the documents, the induction of the decision tree and the keywords extraction of textual homogeneous clusters. The feasibility of the developed model is demonstrated through its applications in three study cases using the CorpusMiner tool. The validation process comprised a linguistic expert's analysis of the obtained keywords and their relation with the topics corresponding to the textual clusters that they characterize.

Keywords: Feature Selection, Decision Trees, Text Mining, Relevance Term.

1 Introduction

A field where feature selection has a significant practical interest is the mining of information, especially, text mining, where the volume of features considered to describe the documents is extremely big and in many cases irrelevant and redundant. Several areas within text mining require a process of feature selection and many techniques of feature selection have been applied to these areas of textual processing; nevertheless, in the majority of cases the efforts in feature selection have been focused in the stage of reduction of dimensionality in textual representation. Nevertheless, there exist many other stages of the textual processing where feature selection becomes necessary. The aim of this work is to apply the feature selection techniques to a textual corpus previously classified to obtain the relevant features that are capable of characterizing the textual clusters and simultaneously manage to discern among clusters.

This paper is organized as follows. In section 2 there will appear a classification of the feature selection techniques and the principal algorithms applied in text mining will be mentioned briefly. The model proposed for the selection of relevant terms in homogeneous clusters of documents is presented in section 3. The evaluation of the

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 137-148

Received 23/02/07
Accepted 08/04/07
Final version 23/04/07

model appears in section 4. Finally, in section 5 the principal conclusions are outlined and possible applications are proposed.

2 Feature Selection in Text Mining

Feature selection that is done in the textual representation uses a filter approach. Thus, the final vocabulary is established by selecting all those features whose score is higher or lower than a predetermined threshold or selecting the best m features. To apply feature selection techniques in textual domains it is necessary preprocess the documents and represent them structurally. One of the representations most widely used in textual domains is the Vector Space Model (VSM) [10].

We will mention only some filter methods to select features used in dimensionality reduction in the textual representation task. A well-known linguistic approach is the stop word elimination [9][12]. Several numerical measurements are frequently used to evaluate the quality of the terms; e.g. eliminate all the terms whose frequencies are either higher or lower than a predefined threshold [9], consider the importance of the terms (term frequency / inverse document frequency (tfidf)) [11], consider the entropy of the probability distribution of the terms among the documents [12] and calculate measurements that are used to calculate the quality of the terms [2]. So far examples of forms of feature selection in the stage of textual representation have been given. Nevertheless, other stages exist in the textual processing that need to apply feature selection techniques, generally those that extract knowledge from texts. For example, if it is desirable to obtain an extract from every obtained cluster as result of a clustering process, it is not possible to consider all the words that were obtained in the process of dimensionality reduction of the VSM, but it becomes necessary to submit every cluster to a new dimensionality reduction process.

3 Model for the Selection of Relevant Terms in Textual Clusters

The objective of the model is to achieve the selection of keywords that characterize homogeneous clusters of related documents and simultaneously manage to discern between the clusters. The input to the model is the result of the documents clustering, where the classes to which every document corresponds are the result of the clustering process and the principal output are the keywords that characterize and discern between the homogeneous clusters of documents. Two secondary outputs, but also of big profit are: the decision tree and the rules of induction.
The relevant terms obtained can be used in later processes such as the extraction of summaries of the multiple documents that compose a homogeneous cluster and in the labeling of the clusters.

As part of the conceptual model a general procedure is developed that includes several specific procedures, structured in three stages that as a whole summarize the content of the model. The stages of the general procedure are (see Fig. 1): (1) discretization of the features that describe the documents, (2) induction of the decision tree, and (3) extraction of keywords from homogeneous textual clusters.

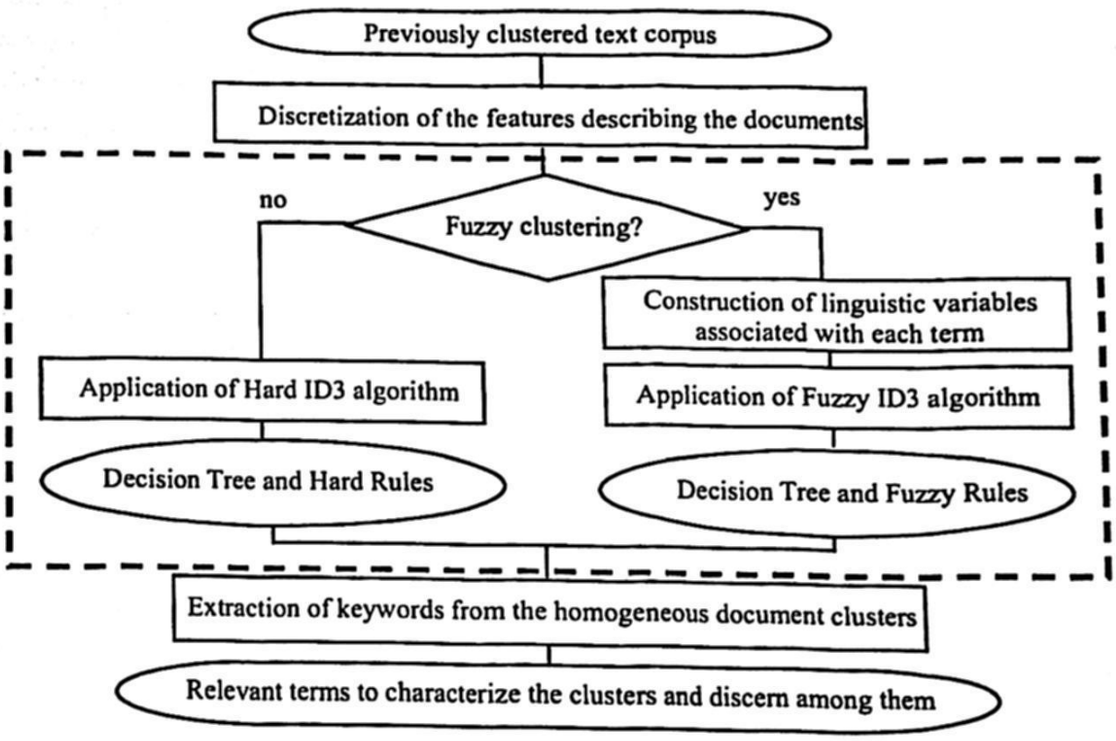


Fig. 1. Model proposed for the selection of relevant terms that characterize textual related clusters and can discern among them.

3.1 Input to the General Procedure

The input to the model is the result of the document clustering, where the classes to which each document corresponds are the resulting clusters of this process. They are considered to be outputs of clustering methods that include any of the three following techniques: hard and deterministic, fuzzy, or hard and overlap [4]. This general procedure is incorporated into the CorpusMiner system [1], that starts from a VSM representation of the document collection, whether modified or not by the application of some technique of normalization, weighting of the matrix elements, reduction of dimensionality or any combination of these, and it clusters the documents following some of these algorithms: Simultaneous Keyword Identification and Clustering of Text Documents (SKWIC) [2], Simultaneous Keyword Identification and Fuzzy Clustering of Text Documents (Fuzzy SKWIC) [2], and Extended Star algorithm [3], or the concatenated variants Extended Star – SKWIC and Extended Star – Fuzzy SKWIC [1].

We represent both input possibilities in Table 1. Clusters for hard ID3 specifies for each document to what cluster it belongs whereas cluster for fuzzy ID3 shows for each document the membership degree to each cluster obtained.

Table 1. Matrix of input to the hard and fuzzy ID3 algorithms.

	T_1	T_2	...	T_m	... for ID3	... for Fuzzy ID3
D_1	$tf_{d_1}(t_1)$	$tf_{d_1}(t_2)$		$tf_{d_1}(t_m)$	C_{k1}	$(\delta_{Cluster1}(D_1), \ldots, \delta_{Clusterk}(D_1))$
D_2	$tf_{d_2}(t_1)$	$tf_{d_2}(t_2)$		$tf_{d_2}(t_m)$	C_{k2}	$(\delta_{Cluster1}(D_2), \ldots, \delta_{Clusterk}(D_2))$
...			...		...	...
D_n	$tf_{d_n}(t_1)$	$tf_{d_n}(t_2)$		$tf_{d_n}(t_m)$	C_{kn}	$(\delta_{Cluster1}(D_n), \ldots, \delta_{Clusterk}(D_n))$

Where D_i is the i-th document of the corpus, with $i \in \{1, n\}$ and T_j is the j-th term that describes the documents, with $j \in \{1, m\}$; the values tf are the normalized and weighted frequency of each term for each document.

3.2 Stage 1: Discretization of the Features that Describe the Documents

The general procedure proposed considers in the third stage the application of the ID3 algorithm, whether in its hard or fuzzy variant. In the first case it is required that the features describing the problem are already discretized, in the second one it becomes necessary to construct the linguistic variables associated with the terms that describe the documents. Thus, stage 1 is so important, whether the processing is begun from the result of a hard or a fuzzy clustering. The proposed procedure considers in this first stage the discretization using the equal width method, that is to say, it partitions the set of possible values of the term in intervals of equal size [5].

3.3 Stage 2: Induction of the Decision Tree

The classic algorithm in the induction of decision trees is the algorithm ID3 for discreet values [6][7] and its extension C4.5 for continuous values [8]. To obtain the induction of the decision tree from the collection of documents (documents that originally are not labeled), it is necessary to consider the result of a clustering process as the set of classes of the document collection. The classification of every document consists of the clusters to which it belongs after the clustering process, or, in the fuzzy variant, its membership degree to each cluster obtained. Thus, the methods ID3 [7], C5.4 [8] and fuzzy ID3 [13] are those which will be used in the selection of the terms that characterize the homogeneous clusters of related documents. The first one is used if the input to the general procedure is the result of an algorithm that applies a hard and deterministic technique and the second one will be used if the technique is fuzzy. When the technique is fuzzy it becomes necessary to obtain the linguistic variables associated with every term that describes the collection.

The fuzzy ID3 algorithm [13] offers the possibility of considering weights associated with the objects of the decision system. In this paper two ways are considered of obtaining information to weight the documents: the highest membership degree of the document to a cluster and the variance of the membership degrees of the document to all the clusters. The highest membership degree of every document to the clusters is a way that is proposed in this paper to weight the documents, because the

higher the membership degree, the more typical or representative the document is of this cluster, therefore it must have a higher responsibility in the induction of the fuzzy decision tree. Those documents that have a similar membership degree for all clusters are documents that are not typical or typical of no cluster in particular, therefore, these documents must have a lesser influence in the selection of the algorithms to induce the fuzzy tree. Hence the documents that have a higher variance of the membership degrees to the clusters must have a major leading role in the induction. Thus, one of the criteria proposed in this paper to weight the documents is to consider the variance of the membership degrees of each document to all the clusters.

In the hard decision trees, it is simple to select which examples belong to the node that is being constructed associated with a value of a certain attribute. On the other hand, fuzzy logic states that all the elements belong to all the sets but with a given membership degree. Therefore, in the FDT after an attribute branches for a certain linguistic term, all the examples have a membership degree to this linguistic term. Let's suppose that there is a node corresponding to the linguistic variable j and that this variable is composed of k linguistic terms. Let's also suppose that the new node to form is that corresponding to the linguistic term a_t^j (linguistic term t of the linguistic variable j). What examples to consider in the node to branch associated to a_t^j ? To solve this problem in this paper two variants are proposed. The first is to apply the Principle of Maximum Membership, so that those examples which have the highest membership degree to that linguistic term t of the linguistic variable j will be included in the corresponding node of a_t^j. The second variant considers the specification of a threshold α. To include an example in the corresponding node of a_t^j, we apply an α- cut and select those examples for which it is fulfilled that its membership degree to the linguistic term t is higher than the α- cut.

Another important aspect to define is when to finish the ramification of the tree. In this stage three stopping criteria have been included. The most general is to stop the ramification when there are no any more attributes or features for the classifications. The second included criterion is to consider a node leaf when all the examples belong to the same class. The third and last criterion considered in the induction compares the value of the measurement of the information of the attribute with a stated threshold, if the studied value is minor that a threshold given by the user or calculated by the proper algorithm, stops the ramification. This stopping criterion avoids choosing attributes with very low information gain. The automated variant calculates the threshold as the average of the information gain of the attributes that initially describe the set of examples.

It is necessary to specify how it is identified to what class or set of classes an example belongs, to determine the classes associated with a node leaf. In the hard variant of the ID3 it is trivial. Nevertheless, in the fuzzy variant of the ID3 all the examples have a membership degree to each of the classes. We propose the use of the following two ways of choosing the classes corresponding to an example or node leaf:

- To apply the Principle of Maximum Membership, in such a way that the only class associated with the example will be that class for which the example has the highest membership degree. To identify the class associated with a node leaf this

principle is applied to every example of the node and the class that has the highest number of associated examples is selected.
- To define a threshold α and to apply an α- cut, thus all those classes are included in the classification of an example for which the example belonged with a degree above the threshold α. To identify the classes associated with a node leaf, an α- cut is applied to every example of the node and the classes associated with this node will be all those obtained from the examples that belong to the node according to this criterion. This is one of the advantages of the FDT, because this method allows a node leaf to have more than one value of the decision attribute.

Another element to bear in mind to obtain a node leaf is the definition of its certainty, an important aspect when generating and applying the rules from the FDT. We have considered two ways of calculating the certainty of a node leaf (certainty of the rule that generates from the root up to the given node leaf). (i) To calculate the certainty of the node leaf as the average of the membership degrees of the examples those are in the node to the classes selected for this node. (ii) To consider the weighted sum of the membership degrees of the existing examples in the node leaf to the classes selected for this node. The weighting is based on the weight associated with every example. In both variants a decision tree is generated and the rules that describe every document cluster are obtained from the generated tree. The antecedents of these rules are the resultant intervals of the discretization process (hard variant) or the terms associated to the linguistic variables (fuzzy variant). The extraction of keywords is performed based on the analysis of the obtained rules. This process will be described in stage 3.

Fuzzy ID3 variant needs the construction of linguistic variables associated with every term. Several researches have been carried out with the purpose of automatically building membership functions. We used the method propose in [12] for the construction of triangular and the Beta bell functions.

Generation of Rules that Describe a Textual Corpus from the Hard and Fuzzy Variants of the Algorithm ID3. After the tree has been constructed the rules that describe the text corpus can be generated, bearing in mind that every path in the decision tree of the root to the leaves is a rule, where the precedent is a conjunction of all the internal nodes of the tree that belong to the path (with its respective discrete associate values or linguistic terms for ID3, hard or fuzzy variant, respectively) and the consequent is the node leaf (i.e., associate classes and certainty of the rule). The rules that are obtained from the tree induced for fuzzy ID3 are Sugeno grade 0.

3.4 Stage 3: Extraction of Keywords of Textual Homogeneous Clusters

In this stage of the model we propose three variants of feature selection to extract the keywords of the homogeneous clusters of related documents obtained from the clustering results. This may be useful, for example, for a possible later stage of automatic generation of the summary extract of every cluster or labeling of textual clusters. Thus we may identify those terms that characterize every cluster, through the selection of the words of higher relevance of clustering methods, the selection of the

terms with higher values of quality in the cluster, and the selection of the terms from the rules generated by the algorithm ID3 in any of its variants.

The selection of the words of higher relevance resulting from the clustering method, and the selection of the terms with higher quality values in the cluster, are forms of selection that coincide when the clustering was performed applying either a hard or a fuzzy technique. The only difference is that when the technique is fuzzy it becomes necessary to apply the principle of maximum membership or to define one α- cut to determine what documents belong to each cluster. Nevertheless, the election of the terms from the rules generated by the algorithm ID3 depends on whether the variant was hard or fuzzy.

Selection of the Words of Higher Resultant Relevancy of the Clustering Methods. There exist clustering algorithms that along with the collection of document clusters return the relevance of the terms for each cluster; such is the case of the algorithms SKWIC and Fuzzy SKWIC. Considering the relevance of the terms for clusters it is possible to select the relevant terms for every obtained cluster in two ways: selection of the words whose relevance is higher than a certain threshold and selection of n words with better relevance value. The first one goes through all the terms in the cluster and chooses those whose relevance value is higher than a stated threshold. The second one sorts all the terms in decreasing order according to their relevance and selects the first n terms of the list. Notice that this way of keyword selection is only applicable to results of the clustering with SKWIC and Fuzzy SKWIC.

Extraction of Keywords as the Quality of Terms. As mentioned in section 2 of this paper, there exist functions that determine the quality of a term in a document collection. Using these quality measurements it is possible to reduce the dimensionality of the VSM representation of a text corpus by eliminating the words of lesser quality value. It is possible to extrapolate this form of selection of words to each cluster of documents obtained by a method of clustering. To achieve this, it is necessary to create a VSM representation for each cluster of documents. This representation has the same terms as the matrix from which the collection of clusters of documents was obtained, but for each cluster the representation has only those documents belonging to it. Then, for each of these representations the quality of all the terms is calculated and those are chosen which have higher or lower value than a threshold (lower in the case of entropy) or the n better quality terms.

Selection of the Terms from the Rules Generated by the Algorithm ID3. From the obtained rules it is possible to generate the words that discern among clusters. This process of selection differs according to whether the rules are hard or fuzzy.

The selection of terms from hard rules is carried out as follows. With every cluster there are associated those terms that are a part of the antecedents of the rules of which that they are consequents and whose value (i.e., interval associated in the process of discretization) is one of the n better values that this term can reach, where n is a value of input to the algorithm. Let's suppose, for example, that one of the terms that describes a textual corpus is the word SOFTWARE, and in the process of

discretization the frequencies of appearance in the corpus for this term were divided into as VERY LOW, LOW, NORMAL, HIGH and VERY HIGH, and it has been specified except that only it is desirable to consider in a cluster those terms that should describe it with a frequency of appearance HIGH or VERY HIGH, then only SOFTWARE will be considered to be a relevant term in those clusters where this term is a part of the antecedents of the rules and that these rules have the above mentioned cluster as a consequent.

The selection of terms from fuzzy rules is carried out as follows. With each cluster are associated those terms that are a part of the antecedents of the rules of which they are consequents and that its value (i.e., linguistic term associated with the linguistic variable corresponding to the attribute that describes the node) is one of n better values that this term can have, where n is a value of input to the algorithm. Therefore, this variant is similar to the processing of hard rules, with the only difference that is processed through linguistic terms and not through discrete intervals.

Combined Forms of Selection. Notice that the ways of selection showed above are used to extract relevant terms for clusters, but these do not necessarily manage to discern among clusters. On the other hand, the use of the rules obtained by the induced decision trees can generate terms that do discern among clusters, but that do not necessarily have a high frequency of appearance in the cluster, and therefore, are terms that can hardly be used in later processing as extraction of summaries or labeling of clusters. It is for these reasons, that combined variants offer the best solutions. If the algorithm that generated the collection of clusters of documents also generated the relevance of every term for cluster, the lists of keywords that are obtained by the ID3 (hard or fuzzy) can be intercepted with the lists of words that are obtained when terms are chosen based on their relevance above a certain threshold. Another possible combination is to intercept the results of the ID3 with the high quality words obtained for each cluster.

4 Evaluation of the Model

Evaluating is an arduous work in tasks of text mining. To evaluate the model study cases were designed and experts' opinion was considered to perform the semantic analysis of the words extracted in the context of the textual corpus used.

The general procedure was implemented in the software CorpusMiner [1]. This software also implements the initial processing and clustering of the textual corpus. In the stage of textual representation, the corpus was transformed by conversion of all the letters to capital letters, replacement of the contractions with their expansions and of abbreviations with their full forms, the elimination of numbers and symbols, the establishment of orthographic homogeneity and lemmatization. Then the VSM representation was carried out on the transformed corpus, with a weighting based on a variation of the formula TF-IDF [2], allowing that the weight of the terms should reflect the relative importance of a term in a document with regard to other terms in the document. The reduction of the dimensionality was performed by the elimination of the grammatical words and the selection of those 600 better terms, that is to say,

terms that have a higher quality than a certain threshold for the applied measurement of term quality (Term Quality II) [1][2]. The concatenated methods applied for the clustering were Extend Star – SKWIC and Extend Star – Fuzzy SKWIC [1][2][3].

4.1 Definition of the Study Cases for the Application of the General Procedure

The first study case included a textual corpus that was created from the Reuters Agency news collection published by David D. Lewis. The second study case is an artificial corpus created by expert linguists for this validation. Finally, the third study case is a collection of Bioinformatics' scientific papers published in the BioMed Central's open access full-text corpus for data mining research.

The first study case includes the textual corpus from Reuters Agency news collection[1]. The created corpus has a size of 353 KB. It possesses 113 pieces of news, previously labeled. These documents tackle 6 topics; 12 news about cocoa, 23 news about acq, 12 news about money-supply, 17 news about trade, 24 news about crude, and 25 news about earn.

The second study considers a textual corpus that was constructed intentionally by expert linguists to validate clustering for meaning association. The construction of this corpus starts from a collection of documents, from which the lexical high frequency words were selected and the sentences containing them were assigned to a document. Thus for each word selected there is a pseudo-text that contains a variable number of sentences containing that word. The built corpus is composed by 35 documents (corresponding to the 35 most frequent words of the original corpus) and it occupies 2.78 MB.

The third study case includes the textual Corpus from BioMed Central's open access full-text corpus[2]. The created corpus has a size of 3.08 MB. It possesses 123 scientific papers, previously labeled. These documents tackle 7 topics; 16 papers about Cystic fibrosis, 12 papers about genic therapy, 6 papers about diabetes mellitus (therapy and diet), 32 papers about diabetes mellitus (research, molecular biology), 31 papers about AID, 16 papers about lung cancer, and 10 papers about microarrays.

4.2 Validation of the Results

To validate the results of the first and second study cases we considered the opinion of experts in English to determine the appropriateness of the keywords selected for the clusters. In their opinion the words obtained manage to describe the clusters and to discern among them.

We apply a fuzzy clustering technique for these study cases; thus, we had to apply the induction of the fuzzy decision trees using fuzzy ID3 algorithm. Tables 2 and 3 reflect fragments of the results of the keyword selection process from the clusters for the first and second study cases. The terms shown are the result of the selection

---

[1] Reuters-21578 Text Categorization Collection, 135 topics.
http://www.daviddlewis.com/resources/testcollections/reuters21578

[2] BioMed Central has so far published 22003 articles.
http://www.biomedcentral.com/info/about/datamining/

method that considers the interception of the terms obtained by fuzzy ID3 and the relevance of the terms calculated by the Extended Star – Fuzzy SKWIC (i.e, we used a fuzzy clustering technique). In the discretization process for the automatic construction of the membership function three frequency intervals were considered, LOW, NORMAL and HIGH. In the selection we only considered those terms that were part of the antecedents associated to linguistic terms with an HIGH frequency and a relevance based on the results of the Extended Star – Fuzzy SKWIC algorithm above a given threshold (which was different for the two study cases). In the induction of the fuzzy decision tree each document was weighted with the variance of the membership degrees of the document to each of the clusters. We applied the principle of maximum membership to determine what examples to include in the branching of the node. The branching of each node was pruned when the attribute gain was smaller than the means of the gains of the attributes that initially describe the set of all examples. The certainty of the rules was calculated as the weighted sum of the examples' membership degrees to the classes selected for this node leaf (classes whose examples' membership degree is above a given α-cut).

Table 2. An excerpt of keywords obtained using Fuzzy ID3 with the first study case.

Cluster 4		Cluster 6		Cluster 5		Cluster 1	
<TRADE> (0.76)		<EARN> (0.72)		<CRUDE> (0.86)		<COCOA> (0.91)	
System	important	dollar	Exchange	military	risk	Cocoa	bean
Foreign	export	represent	Pay	platform	involve	Strong	cost
Tender	trade	cause	Earn	Washington	predict	Carry	rise
Sight	spokesman	financial	Asset	target		Land	dollar
Gulf	congress	partly	intangible	Iran		Pound	condition
		asset				Offer	good

The headings for each cluster in Tables 2 and 3 are the topics most widespread in the cluster and their degrees of importance. Observe in Table 3 that the chosen terms have a relationship with the main topics dealt with in these homogeneous document clusters, so the semantics of the selection is correct and adequately describes the clusters under study. These words also discern among the clusters.

See in Table 3 the results of the keywords obtained for a fragment of the clusters of the collection of the second study case. Notice that the chosen terms are in correspondence with the content of the clusters.

Table 3. An excerpt of keywords obtained using Fuzzy ID3 with the second study case.

Cluster 1	Cluster 2	Cluster 4	Cluster 5	Cluster 6
<virtual> (0.89)	<wireless> (0.77) <users> (0.74)	<virus> (0.84)	<woman > (0.83)	<warming> (0.90) <weather> (0.83)
virtual	wireless	virus	woman	weather
system	user	system	female	warm
future	computer	seriously	men	troposphere

To validate the results of the third study case, we considered the opinion of experts in Bioinformatics to determine the appropriateness of the keywords selected for the clusters of this kind of scientific papers. In their opinion the words obtained manage to describe the clusters and to discern among them.

We decided to use a hard and deterministic clustering technique; thus we needed to apply Hard ID3 to induce the hard decision tree. Table 4 reflects the complete results of the keyword selection process from the homogeneous clusters of Bioinformatics' papers. The keywords shown are the result of the selection method that considers the interception of the terms obtained by ID3 and the relevance of the terms calculated by the Extended Star – SKWIC (i.e, we used a hard and deterministic clustering technique). In the discretization process three frequency intervals were considered, LOW, NORMAL and HIGH. In the selection we only considered those terms that were part of the antecedents associated to intervals with a HIGH frequency and a relevance based on the results of the Extended Star –SKWIC algorithm above a given threshold. The threshold is given considering the characteristics of this study case.

Table 4. The keywords obtained using ID3 with the third study case.

Cystic Fibrosis	Genetic Therapy	Diabetes (Diet and therapy)	Diabetes (Research)	AID	Lung Cancer	Microarray
cystic	Virus	people	experiment	HIV	response	clone
fibrosis	Transfer	prevention	cell	program	tumor	microarray
surface	DNA	primary	clone	human	Lung	experiment
line	Joint	model	pain	prevention	clone	
	AAV	status	mouse	transmission	DNA	
	Tumor		human	information		
				health		

See in Table 4 that the chosen terms have a relationship with the main topics dealt with in these homogeneous document clusters, so the semantics of the selection is correct and adequately describes the clusters under study. These words also discern among the clusters.

5 Conclusions and Future Work

A model has been presented that allows the selection of keywords that characterize homogeneous clusters of documents and can simultaneously discern among the clusters. The characteristics present in the general procedure of the developed model provide advantages with regard to the consideration in the input of new ways of document clustering and the inclusion of other variants construction of membership functions associated to the linguistic variables for every feature. The stage of keyword extraction allows the combination of the relevancy of the features obtained by the clustering processes, with the words selected from the process of induction of the hard or fuzzy decision trees. This element is fundamental to obtain relevancy for clusters and differentiation among them.

As a future work, it is possible use the relevant terms obtained in the extract summarization of textual homogeneous clusters, as well as in the labeling clustering process.

Acknowledgments. This work was supported in part by VLIR (Vlaamse Inter Universitaire Raad, Flemish Interuniversity Council, Belgium) under the IUC Program VLIR-UCLV and by Informatics for Enterprises Project between Cuba and Germany. Thanks also to Prof. Rudolf Kruse, Prof. Christian Borgelt and Prof. Andreas Nuernberger for their suggestions.

References

1. Arco, L., Bello, R., Mederos, J.M., Pérez, Y.: Agrupamiento de documentos textuales mediante métodos concatenados. Revista Iberoamericana de Inteligencia Artificial, 10(30) (2006) 43-53
2. Berry, M.: Survey of text mining. Clustering, classification, and retrieval. Springer-Verlag. (2004)
3. Gil-García, R., Badía-Contelles, J.M., Pons-Porrata, A.: Extended star clustering algorithm. Proceedings of CIARP. Lecture Notes in Computer Science, 2905, Springer-Verlag (2003) 480-487
4. Höppner, F., Klawonn, F., Kruse, R., Runkler, T.: Fuzzy cluster analysis. Methods for classification. Data Analysis and Image Recognition. John Wiley & Sons Ltd. (1999)
5. Liu, H., Setiono, R.: Feature selection via discretization. IEEE Transactions on Knowledge and Data Engineering, 9(4) (1997) 642-645
6. Mitchell, T.: Machine learning. McGraw-Hill Science (1997)
7. Quinlan, J.R.: C4.5: Programs for machine learning. Morgan Kaufmann Series in Machine Learning (1993)
8. Quinlan, J.R. Improved use of continuous attributes in C4.5. Journal of Artificial Intelligence Research, 4 (1996) 77–90
9. Rijsbergen, C.J.: Information Retrieval. London, Butterworths (1979)
10. Salton, G., Wong, A., Yang, C.S.: A vector space model for automatic text retrieval. Communications of the ACM, 18(11) (1975) 613-620
11. Salton, G., Buckley, C.: Term weighting approaches in automatic text retrieval. Information Processing and Management 24(5) (1988) 513-523
12. Salton, G., McGill, M.: Introduction to modern information retrieval. New York: McGraw-Hill (1983)
13. Wang, X., Borgelt, C.: Information measures in fuzzy decision trees. IEEE International Conference on Fuzzy Systems (2003)

Design of an Explicative Combinatory Dictionary for Spanish Verbs

Gaspar Ramírez[1], Jim Fidelholtz[1], Héctor Jiménez-Salazar[2]

[1] Posgrado en Ciencias del Lenguaje,
B. Universidad Autónoma de Puebla,
Puebla, 72000

[2] Departamento de Tecnologías de la Información
Universidad Autónoma Metropolitana,
Cuajimalpa, 11850
{gasparfirst, fidelholtz}@gmail.com, hgimenezs@gmail.com

Abstract. Several lexical resources have been proposed in the literature. However, the main problems of Natural Language Processing (NLP), such as ambiguity resolution and parsing, require a more integrated version of the different approaches, i.e. an organized, theoretically well-motivated form of lexical, syntactic and semantic information for each entry. In this paper we present the design of a dictionary based on a proposal by Apresjan. We describe the definition zones of entries in the lexicon, the methodology followed to obtain the content of this zones and we give an example in the semantic domain of creation verbs. We also discuss the complexity of carrying out this task.

1 Introduction

Assignment of senses to verbs is not a trivial matter. In modern linguistics and computational linguistics most studies have concentrated on identifying the thematic roles of verbs in order to characterize semantic frames. They also describe how semantic roles can be mapped onto syntax in a predictable way. Likewise, an Explanatory Combinatorial Dictionary (ECD) describes the logical arguments of verbs and the correspondences between semantic and syntactic arguments very adequately, identifying different senses of verbs by enumerating them as sense_1,..., sense_n. We know very little about, among other things, what characterizes sense_1 as distinct from the others and what are the differences and similarities between sense1 and the other senses.

It is not sufficient to enumerate the different senses that a verb can have and acquire based only on semantic features (semantic structure). More important, the semantic structure only deals with the logical meaning of a sentence. Because of this fact, we must also attempt to classify senses according to their cognitive aspects, logical emphasis, presuposedness, social associations which are considered as part of the communicative structure of a sentence within its semantic representation structure. Thus, in this paper the meaning of a sentence

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 149-159

Received 24/02/07
Accepted 08/04/07
Final version 23/04/07

is based on Mel'čuk sentence representation structures [9]. That is, a sentence representation is conformed by a semantic structure, syntactic structure and a communicative structure. Nonetheless, due to some specific phenomena observed within sentences, we also think that it is necessary to include a referential structure to achieve an even broader sentence meaning representation. In order to realize this goal, we have implemented some criteria for establishing senses from a referential view point. Here, a referential structure is considered as specifying the referent of a semanteme configuration. That is, it indicates the links between semantic chunks and the corresponding entities and facts in the universe to which these chunks refer (Mel'čuk, 2001). Then, in order to deal with the referential structure, we assign different tags to different senses to set up the categories in which they participate.

In this proposal, therefore, a standard entry is organized basically into four definition zones:(1) the lemma zone; (2) the explanation and meaning zone; (3) syntax; and (4) the co-occurrence restrictions zone. Since referential structure is one important characteristic of this proposal when classifying senses, it is intended mainly to establish a methodology capable of, on the one hand, identifying all possible senses that a verb can have and, on the other hand, classifying senses into categories by tagging them in a reasonable manner. This will be done by the identification and classification of all different possible senses. This is based primarily on two criteria: (a) the participants (actants) of the verb and (b) the proper features of the verb. Consider the following sentences:

(1) a. Maria rompió la ventana. *Maria broke the window.*
b. El martillo rompió la ventana. *The hammer broke the window.*
c. La ventana se rompió. *The window broke.*

According to Case Grammar [4] [13] and from a communicative point of view these sentences express different senses since sentence (1a) expresses the subject who carries out the action; sentence (1b), however, it expresses the instrument with which the action was carried out; and finally, sentence (1c) shows the object affected by the action of *romper* (break). That is, thematically each one expresses a different theme and rheme due to well-known meaning communicative organization rules. Consequently, these sentences, (1a), (1b) and (1c), should be interpreted differently since they evoke different senses. As we can see, they clearly express different states of affairs in the world. That is the message is communicatively organized in such a way that it is making reference to different things in the world. On the other side, it seems syntactically that the above sentences are composed according to the grammatical constructions SVO (1a and 1b) or SV (1c). However, their semantic meaning has been strongly affected by the communicative structure of the sentences, proving that communicative structure has a great influence in the semantic interpretation. It is quite clear that they do not mean the same thing, since their interpretation depends too on what kind of argument the verb is interacting with, that is, whatever each participant denotes affects the possible meaning that a verb can acquire syntagmatically and paradigmatically.

Based on these considerations and the two criteria mentioned above, it is therefore assumed that identification of what each participant denotes is of great importance in the task of identifying and classifying the different senses of a given verb and arranging senses in different categories. For instance, in the case of the sentences mentioned previously, the meaning of sentence (1b) would be classified under the tag Instrument, Operative Part. Similarly, sentences (1a) and (1b) will be placed under the tag Object/Content, making it clear that (1a) is intended to describe the subject of the sentence, whereas (1c) is making reference to the direct object of the sentence which turns out to be the actant to be affected by the action of the verb. The other important criterion for identifying and eventually classifying all possible senses of a verb is the proper features of the verb examined. As an example, the verbs *comer* (eat) and *devorar* (devour), which are normally considered instances of the verb *ingerir* (ingest). From this perspective the purpose is to graduate the meaning of comer and devorar as two distinct manners of ingesting something. Although gradient distinctions are important for semantics, aspect features must be included in order to identify senses more accurately and classify them broadly. Let us consider the following sentences which contain the verbs *comer* (eat) and *devorar* (devour):

(1) d. María comió su torta ayer. *Maria ate her sandwich yesterday.*
e. María comió ayer. *Maria ate yesterday.*
f. María devoró su torta ayer. *Maria devoured her sandwich yesterday.*
g. *María devoró ayer. **Maria devoured yesterday.*

In sentence (1d) and (1e), comer denotes an action of unbounded duration, that is, the sentence does not provide information about the duration of the action, although we know that it has finished, at least deictically, at some time in the past. However, sentences (1f) and (1g), in contrast to (1d) and (1e), make reference to a change of state. This change of state seems to be absent in comer; in other words, in devorar there is a completive implication which is absent in comer. This distinction seems to help us explain the syntactic behavior of these verbs, that is, why comer accepts the dropping of the direct object while devorar does not.

Once all possible senses of a verb have been established and assigned according to its participants and proper features, we determine in which semantic frames verbs participate. Thus we are trying to explain syntagmatically why verb classes behave as they do, and what consequences these distinctions have for the rest of the lexicon and grammar. Consequently, the aim of this research project is to identify compositionally the similarities and differences between verbs, whether or not they are semantic.

The article will be organized in the following manner, we explain in section 2 the organization of semantic classes. In section 3 the structure and composition of verb descriptions are reviewed. Finally, in section four we describe the methodology used and in section five we show what an actual (partial) dictionary entry looks like.

2 Semantic Classes

This project is based on one of the oldest semantic classifications of verbs, their aspectual class or Aktionsart. This classification considers that verbs and verbal phrases differ in the types of events that they denote in the world. In other words, in the kind of action verbs denote. It is normally assumed that there are at least three aspectual types: state, activity, and event, where the last sometimes is divided into accomplishment and achievement events.

Some examples will show what we mean by aspectual class. The verb *caminar* 'walk' in sentence (2) denotes an activity of unbounded duration; that is, the sentence itself does not carry information about the temporal extension of the activity, although deictically it is an event that finished in the past.

(2) María caminó ayer. *Maria walked yesterday.*
(3) María caminó a su casa ayer. *Maria walked home yesterday.*

Thus sentence (2) denotes an activity or an achievement. Other examples of this class of verbs are: *dormir* 'sleep', *correr* 'run', *trabajar* 'work', *beber* 'drink', etc. On the other hand, sentence (3) conveys the same information as (2), except that (3) is constrained by the fact that María finished walking when she arrived home. Although there is no explicit reference to the duration of the activity, this sentence states that the process has a logical culmination, since it finishes when María gets home. It is said that this kind of sentence denotes an accomplishment.

Just as the verb *caminar* seems by default to be an activity in lexical terms, there are verbs that seem to denote accomplishments lexically. For example, the verbs *construir* 'build' and *destruir* 'destroy', in their typical transitive use, denote accomplishment events since there is a logical culmination to the activity performed.

(4) María construyó una casa. *Maria built a house.*
(5) María destruyó la mesa. *Maria destroyed the table.*

In sentence (4) the coming into existence of the house is the culmination of Marías action, while in (5) the nonexistence of something referred to as a table is the direct culmination or consequence of the action of the verb. Verbs of creation are the best examples of accomplishment events.

An achievement, on the other hand, is an event where something undergoes a change of state, as in the case of an accomplishment event, but where the change is conceived of as occurring instantaneously. For example, in sentences (6), (7) and (8) the change is not gradual, but something that has a point-like characteristic to it. Therefore, modification with punctual adverbials such as *a las 3 en punto* suggests that the sentence denotes an achievement event.

(6) Juan murió a las 3 en punto. *Juan died at 3 o'clock sharp.*
(7) Juan encontró su cartera a las 3 en punto. *Juan found his wallet at 3 o'clock sharp.*
(8) Maria llegó a la media noche. *Maria arrived at midnight.*

It seems that some lexical properties of verbs may be affected by the sort of complement with which they interact.

As we can see from the examples so far discussed, the kind of event that a verb denotes may vary from a compositional perspective. Therefore, co-occurrence meaning as well as compositionality should be considered when describing a lexical item. A shift of meaning in the verb arises as a result of syntagmatic interactions and the semantic and syntactic relationship of the verb with the rest of the items in the sentence.

3 The Structure and Composition of Verb Descriptions

As mentioned in the Introduction, the aim of this research project is to identify similarities and differences between verbs, whether or not they are semantic. In this study we use Apresjan's [1] criteria for the elaboration of the *New explanatory dictionary of Russian synonyms*. Here a complete semantic description of a word or collocation consists in an explanation of the meaning of the word (verb, noun, etc.) or collocation (*dar inicio* 'begin') and its meaning. The meaning is also divided into three different categories, namely, purely denotative differences, connotations and, finally, information about its logical emphasis. Thus the composition of a standard entry for a verb is organized in the following definition zones: (1) lemma, (2) explanation and meaning, (3) syntax and (4) combinations and their co-occurrence restrictions (collocations).

3.1 The Lemma

The lemma includes a description of the type of category for each verb. That is, here we mention which category such and such a verb belongs to, the gender, number, etc., of its arguments, etc.

3.2 Explanation and Meaning

Explanation In this study, the explanation of the meaning of a word has two components: (1) the concept that corresponds to the thing, to the situation, to some property, to a state or a process, etc.; that is, what the word denotes. So the concept is described as the frame of the event, which at the same time constitutes an event structure, as well as a sub-event structure. Here also it is important to mention that the concept of a situation (that is, the participants, their characteristics, the relationships among them) can include details about the role which the observer plays. There are also some meanings (and, in consequence, some explanations) which have two different components, namely, presuppositions and assertions. Presuppositions are the elements in an explanation which are not affected by negation. As an example, in the word *soltero* (='bachelor', an unmarried male person who is old enough to marry), the definition element 'person' is a presupposition of *soltero*, while 'unmarried' is an assertion, that is, it is a semantic component which is affected by negation. And (2) the evaluation or

valuation, that is, the opinion of the interlocutors, the speaker or the hearer(s), about the thing, the situation, etc., as good or bad, credible or not, desirable or undesirable, etc. This component makes up the so-called modal framework of the explanation. The description of the concept is the central part of the explanation and the description of the evaluation is its modal framework.

Meaning A rational approach to the different types of meanings of a verbal lexeme is based on the fact that the different lexical entries for the verbs in the dictionary principally denote actions, situations, events, processes, states and properties. The different senses which a verb has may be distinguished both (a) by its participants (subject, object, recipient, instrument, means or vehicle) and (b) by its syntactic-semantic features (cause, effect, purpose, motivation, place, beginning point, end point, time, method of execution, nature, degree, form of manifestation, etc.) Apresjan [1] notes something very important about synonym dictionaries when he says that the types of participants and the generalized features of verbs are what in the majority of cases give us the distinctive features of each verbal lexical item. Thus there have been established purely semantic distinctions, distinctions in logical emphasis and their cultural associations or connotations in order to assign the different senses of a verb.

3.3 Syntax

Verb Alternations We also employ a recently-developed methodology to group the meaning of verbs in semantic classes through the analysis of the syntactic frames in which they participate; that is, common grammatical verb alternations. Here, as an example of similar works, we can mention the MIT Lexicon Project [7], which outlines a large classification of argument-verb alternations in English in order to classify verbs into semantically unique classes.

Let us consider the following examples, the verbs *hundir* 'sink', *rodar* 'roll', and *romper* 'break' all have transitive and intransitive forms when their lexical senses are related to the interpretative characteristic of causation[3].

[3] In (9a), the pragmatically unmarked supposition is that the weather caused the sinking, while in (9b) the cause of the sinking is explicitly specified, the weather being only a commentary on the event.

(9)	a. El bote se hundió en un clima tormentoso.	*The boat sank in a storm.*
	b. El avión hundió el bote en un clima tormentoso.	*The airplane sank the boat in a storm.*
(10)	a. La pelota rodó por la colina.	*The ball rolled down the hill.*
	b. Bill rodó la pelota por la colina.	*Bill rolled the ball down the hill.*
(11)	a. Súbitamente, la botella se rompió.	*Suddenly, the bottle broke.*
	b. Súbitamente, Maria rompió la botella.	*Suddenly, Maria broke the bottle.*
(12)	a. La carta llegó a tiempo.	*The letter arrived on time.*
	b. *El cartero llegó la carta a tiempo.	**The mailman arrived the letter on time.*
(13)	a. Mi terminal murió anoche.	*My terminal died last night.*
	b. *La tormenta murió mi terminal anoche.	**The storm died my terminal last night.*
(14)	a. La torre de bloc cayó.	*The block tower fell.*
	b. *Zacarías cayó la torre de bloc.	**Zachary felled the block tower.*

Although sentences (12b), (13b), and (14b) are ill-formed, they are certainly understandable. A lexical semantic theory should specify what these classes share: for example, both have intransitive grammatical forms. Thus, it is important to identify similarities among verbs for establishing a domain where lexical items are somehow unified (unification), but equally important is the characterization of how verbs differ (individualization). For example, the latter group (the b. sentences) does not allow the transitive form of the sentence. The question is whether it is possible to identify the linguistically relevant features that lead to the distinct behavior of the transitive verbs above. However, as Pustejosvky [13] claimed, we must realize that explaining the behavior of a verb's semantic class can come about only by acknowledging that the syntactic patterns in an alternation are not independent of the information carried by the arguments characterized in those very patterns. In other words, the diversity of complement types that a verb or other category may take is in large part also determined by the semantics of its complements.

3.4 Collocations

It is worth pointing out that in this work collocations or semi-phrasemes are very important, since they contain a great deal of subjective communicative information. A collocation is a more or less fixed expression which is composed of (at least) two constituents. The meaning of the whole collocation includes the meaning of one of its components but not the other, at least not wholly. For example, *sostener una tesis* 'maintain a thesis', where the phrase is about a thesis but *sostener* 'maintain' does not convey its usual meaning (taken from [10]:269). In addition, Mel'čuk, Clas & Polguère [11] present the Lexical Functions (LF) in their Meaning-Text Theory. According to Corpas Pastor [14] the LF were

proposed as an attempt to systematize all the semantic links that could be established within word combinations, that is, between the base and the collocate. In other words, the lexical functions describe the co-occurrence restrictions of words, as well as the phenomena of derivation. Therefore, the notions of base and collocate are similar to the concepts of key (the word which is applied to determine a lexical function) and the value (the resulting word(s)). For example, applying the lexical function **Realization** to the key *promesa* 'promise' we obtain the value *cumplir* 'fulfill'.

4 Methodology

We used three different linguistic resources (dictionaries) to determine the lemma zone, as well as the explanation and meaning zones. These were: the *Diccionario enciclopédico Larrousse* [8], *the Diccionario Enciclopédico Océano* [12], and the Davies corpus for Spanish [3]. Dictionaries are valuable sources of useful information since they list all senses that a verb can have (or all considered necessary by the lexicographers). This knowledge can be verified using a corpus in order to corroborate the senses described in the dictionaries or to find another possible context in which it is employed in a new way which the listed senses do not cover and which consequently is indicative of a new sense[4].

The next step was, for each verb, to examine the dictionaries in order to ascertain its distinct senses. We then analyzed the Spanish corpus of Davies [3] to match each occurrence of the verb with the different senses of the term described in both dictionaries. Different contexts and participants were considered to match the distinct senses of a given verb in the dictionaries when the use of the examples in the corpus assumes such situations.

In our proposal, different senses that a term can have are not arranged in the traditional way of listing the various senses as $\text{sense}_1 \ldots, \text{sense}_n$; we rather seek an arrangement in accordance with what each different semantic argument or semantic actant denotes. In other words, what type of argument a verb can combine with and what each argument makes reference to are of great importance for compositional interpretation, on the one hand, and for the understanding of co-occurrence phenomena, on the other. In addition, the contribution of the arguments of a verb to its meaning and interpretation is regarded as another important characteristic which distinguishes this proposal.

The steps we have sketched for the procedure of constructing an entry in the ECD, apart from enriching our conception of the content and description of the ECD, permit us to have a better idea of the complexity of this task. It requires the use of linguistic resources (for example, machine readable dictionaries), the tools of NLP must be well-used [2][5][6], and a team of various specialists is needed. The procedures implicit in the construction of an ECD are variegated

[4] Note that not all new collocations imply new senses: in about 1950 we would have found the collocation *prendí/encendí/conecté el radio/la televisión* but hardly the first element with *la computadora*, since the latter was not yet in widespread use; by 1995, all these collocations would have been found in large numbers.

and complex; just to mention a few, we need to make use of parsing, semantic tagging, word sense disambiguation, the identification of collocations and their collocates, and sense clustering.

5 Example Entry

In this work, it was decided to take a specific domain to demonstrate the dictionary entries: verbs of creation. These fall into the more general category of verbs which denote actions and events.

As a rule, both of the dictionaries we used coincided in the number and the description of the different senses of a term; nevertheless, the Larousse dictionary [8] is characterized by its richer lexical semantic descriptions, since it considers a greater number of different contexts for some terms.

The systematic lexicographic description of each verb was organized in the following way. On the one hand, the component of the explanation in which what each verb denotes is established and described, along with the presuppositions and assertions which can accompany certain explanations, and likewise the evaluative frame (evaluation) for each verb. On the other hand, we have the component of the meaning of a verb, in which the different senses which a verb may have are explained, placing these different senses in three separate definition zones: a) their purely semantic distinctions, b) their connotations and c) their logical emphasis. In Figure 1[5] appear each one of the components which are found, for the verb *construir*. Here we must remark that not all the definition zones will always occur for every verb, for example, *edificar* would have the connotation zone filled, while for the verb *construir* is not present. The two components, together with their respective frames and zones, are an attempt to achieve a perspicuous description of the semantic and conceptual organization of Spanish verbs.

Figure 1 shows an example (in part) of what the previous descriptions of our approach lead to.

6 Conclusions

We hope to have presented a cogent case for the usefulness and theoretical desirability of an Explanatory Combinatorial Dictionary for Spanish, as well as an actual partial entry in this dictionary for a member of a certain class of verbs (verbs of creation) in Spanish.

In the example entry, we experimented with a part of the process required for producing entries in the Explicative Combinatory Dictionary of Spanish verbs. Some important points are: (1) the conception of the situation (that is, the participants, their characteristics and the relationships between them) may include details about the role which the observer plays and (2) distinct senses emerge, depending on the combinations of the verbs with their possible arguments within a particular context. With respect to the computational assistance

[5] GL stands for Generative Lexicon [13].

Fig. 1. Combined CED and GL.

Construir. (*Construct*)	v. tr. Del latín *construire*.
Explanation	Acomplishment event
Evento substructure:	E1 < Process
	E2 < State
Presupositions y assertions:	• Presupposes the termination of something.
	• Presupposes a plan.
	• Presupposes grouping the parts to form a whole.
Evaluation:	(+) desirable, (+) Aesthetics (pretty) (+) Ethics (well done)

Meaning
Construir. Hacer una cosa material o inmaterial ordenando y juntando sus partes de acuerdo a un plan.

Object
El objeto sintáctico de construir puede combinar con objetos físicos. Los sujetos por su parte refieren a seres animados y abstracciones de organizaciones sociales (compañías, instituciones). El verbo refiere la existencia de un bien material.
- Juan construyó la casa.
- La compañía construyó el puente después de la guerra.

Content
El objeto de *construir* también denota objetos abstractos. Aquí los sujetos refieren a seres humanos primordialmente. El verbo denota el orden y relaciones del objeto.
- Juan construyó una teoría.
- Construyeron su idea a partir de una fotografía.

Pseudo-passive with *se*[6]
En la cadena comunicativa el verbo *construir* hace referencia a la importancia, relevancia o interés que el objeto tiene sobre el sujeto lo que provoca que aparezca como sujeto gramatical, el sujeto semántico ('subyacente') queda sin especificar. Los sujetos gramaticales son bienes materiales o pueden ser abstracciones.
- El edificio se construyó en 1991.

Instrument
El verbo *construir* puede expresar también el instrumento con el cual se lleva a cabo la construcción de la obra como el sujeto gramatical. En este caso los argumentos con los que interactúa señalan herramientas tales como maquinaria y distintos dispositivos tecnológicos.
- La máquina construyó la carretera.

Definition zones of the verb *construir*.

$$
\left[\begin{array}{lll}
\text{Construir} & & [\textit{Construct}] \\
\text{EVENTRSTR} = & \left[\begin{array}{ll} \text{E1=} & \textbf{process} \\ \text{E2=} & \textbf{state} \\ \text{REST=} & \leftarrow \\ \text{HEAD=} & e_1 \end{array}\right] & \\
\text{ARGSTR} = & \left[\begin{array}{ll} \text{ARG1=} & 1\left[\begin{array}{l}\textbf{animate/inanimate-individual} \\ \text{FORMAL=}\textbf{physobj/absobj}\end{array}\right] \\ X=1; & \text{who constructs?} \\ \%\ \textit{Juan} \sim & [\%\ \textit{John} \sim] \\ \text{ARG2=} & 2\left[\begin{array}{ll}\textbf{entity} & \\ \text{CONST=} & 3 \\ \text{FORMAL=} & \textbf{physobj/absobj}\end{array}\right] \\ Y=2; & \text{what?} \\ \%\sim \textit{una silla} \simeq & [\%\sim \textit{a chair} \simeq] \\ \%\sim \textit{una teoría} \simeq & [\%\sim \textit{a theory} \simeq \\ \text{D-ARG1} & 3\left[\begin{array}{ll}\textbf{material} & \\ \text{FORMAL=} & \textbf{mass}\end{array}\right] \\ Z=3; & \text{from what?} \\ \%\simeq \textit{de madera} & [\%\simeq \textit{of wood}] \\ \%\simeq \textit{sobre el clima} & [\%\simeq \textit{about the climate}] \end{array}\right] & \\
\text{QUALIA} = & \left[\begin{array}{ll} \textbf{create-lcp} & \\ \text{FORMAL=} & \textbf{exist}(e_2, 2) \\ \text{AGENTIVE=} & \textbf{build-act}(e_1, 1, 3) \end{array}\right] &
\end{array}\right]
$$

for the development of this project, we think that, at this stage of the project, the identification of verbal arguments, through the use of concordances, and their semantic classification would be of great assistance to the lexicographer.

Compiling the dictionary following the methodology proposed, we observed that apparently differences and similarities among verb senses do not depend exclusively on prototypical aspects; rather an schemata seems to arise which describes what senses are shared or not by two verbs considered synonyms. That is, two verbs may vary in accordance with their differences and similarities but within a schema. Though at the moment this proposal of dictionary is oriented for humans use, we think that the different zones can be codified in such a way that an automatic system can easily manage them and exploit them in NLP tasks. In addition, in future works the matrix for each verb description will include transformations (transmutations) so that each definition zone (consequently, meaning) is mapped onto its corresponding syntax representation.

References

1. Apresjan, Jurij. 2000. Systematic lexicography. Oxford University Press, Oxford.
2. Bolshakov, I.; Gelbukh, A. & Galicia Haro, S. 1999. Electronic Dictionaries: For both Humans and Computers. J. International Forum on Information and Documentation, N 3, pp. 1722; http://nlp.cic.ipn.mx/Publications/1999/Forum-Dicts-eng.htm.
3. Davies, Mark. 2002. Corpus del español. (`www.corpusdelespanol.org`).
4. Fillmore, Charles. 1968. The case for case. In Universals in linguistic theory, ed. by Emmon Bach and Robert Harms. New York: Holt, Rinehart, and Winston.
5. Galicia Haro, S.; Bolshakov, I. & Gelbukh A. 1998. Diccionario de patrones de manejo sintáctico para análisis de textos en español. J. Procesamiento de Lenguaje Natural, No 23, pp. 171176.
6. Gelbukh, A. and Sidorov, G. 2003. Hacia la verificación de diccionarios explicativos asistida por computadora, in *Estudios de Lingüística Aplicada*, 21 (38), pp 89-108.
7. Hale, Ken and S. J. Keyser. 1986. Some transitivity alternations in English. (Lexicon project working papers 7.) Cambridge, MA: MIT. Center for Cognitive Science.
8. Larousse. 2000. Diccionario enciclopédico Larrousse. México, D. F.: Larousse.
9. Mel'čuk, Igor A. 2001 Communicative organization in natural language: the semantic-communicative structure of sentences. Amsterdam: John Benjamins.
10. Mel'chuk Igor A. 1988. Dependency syntax: theory and practice. Albany, NY: State University of New York Press.
11. Mel'chuk I. A; A. Clas ; & A. Polguère. 1995. Introduction à la lexicologie explicative et combinatoire. Paris/Louvain-la-Neuve: Duculot.
12. Océano. 1995. Diccionario enciclopédico Océano. México, D. F.: Océano.
13. Pustejovsky, James. 1995. The generative lexicon. Cambridge, MA: The MIT Press. Fidelholtz. 2006. Towards a in Computer Science, 107-112. México, DF: CIC-IPN.
14. Corpas Pastor, Gloria. 2003. Diez años de investigaciones en fraseología: análisis sintáctico-semánticos, contrastivos y traductológicos. (Lingüística Iberoamericana 20.) Madrid: Iberoamericana.

On the Use of Dynamic Information for Speaker Identification

Rosa M. Ortega-Mendoza, Esaú Villatoro-Tello, Luis Villaseñor-Pineda, Manuel Montes-y-Gómez and Eduardo F. Morales

Language Technologies Group, Computer Science Department,
National Institute of Astrophysics, Optics and Electronics (INAOE), Mexico
{rmortega, villatoroe, villasen, mmontesg, emorales}@inaoep.mx

Abstract. This paper describes a language independent method for speaker identification. This method is based on a novel characterization of the speech signal that captures the dynamic information contained in the cepstral coefficients. The proposed method was evaluated through several experiments on a corpus of Mexican speakers. The achieved results demonstrated the relevance of the signal characterization, reaching an identification accuracy as high as 97% under a multi-class scheme.

1 Introduction

The speech signal works as a vehicle for several types of information. It mainly bears a message through the language, but it also allows identifying the spoken language and establishing the emotion, the gender, the age as well as the identity of speakers.

In the task of automatic speaker recognition focuses on determining the identity of speakers through their voice. It involves two kinds of problems, namely, speaker identification and speaker verification. Speaker identification, the subject of this paper, determines which registered speaker provides a given utterance from amongst a set of known speakers. On the other hand, speaker verification accepts or rejects the identity claim of a speaker.

Speaker identification has been widely studied and it is currently performed at very high accuracy rates [1]. However, the best achieved results correspond to methods that are text dependent (users must read a predefined text) and language dependent (they depend on the availability of a phonetic recognizer) [2, 3].

In order to diminish these limitations, this paper proposes a new statistic method for speaker identification. This method is *text independent* [4], and also *language independent*. It directly works with the speech signal and avoids applying any process for phonetic recognition. In particular, the proposed method takes advantage from the dynamic information contained in the Mel Frequency Cepstral Coefficients to enhance the characterization of the speech signal.

The rest of the paper is organized as follows. Section 2 describes the proposed method. Section 3 presents the experimental results. Finally, section 4 depicts our conclusions and future work.

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 161-166

Received 23/02/07
Accepted 08/04/07
Final version 22/04/07

2 Proposed Method

The speaker identification task consists of two main phases: a training phase and a test phase. In the training phase an identification model is built from the recordings of every user. Subsequently, in the test phase, these models are used to determine the corresponding speaker for an input recording.

The construction of the identification models involves two processes. On the one hand, the characterization of the speech signals by the extraction of some descriptive parameters. On the other hand, the application of a stochastic procedure –over the extracted characterizations– in order to capture the distinctive regularities for each speaker. Traditionally, a cepstral analysis has been used for characterizing the signals, and Gaussian Mixture Models for inducing the identification models [4].

The contribution of the proposed method is on the application of a different speech characterization. It uses the cepstral coefficients to compute a new set of features that capture the time variations of the signals. These new features resume the dynamic behavior of the signals and thus enhance the construction of the speaker's models. In addition, our method accomplishes the statistical modeling by using some automatic classification algorithms, in particular, Naïve Bayes and Support Vector Machines.

The following sections describe in detail the proposed approach.

2.1 Signal Characterization

In order to construct the speech characterization the signal is represented by fixed-size segments and each segment is characterized using the Mel Frequency Cepstral Coefficients (MFCC). Basically, we consider non-overlapping segments of 30ms and calculate 16 coefficients per segment. We propose using 16 coefficients, instead of the twelve traditionally used for speech recognition, because we want to exploit all possible useful information to distinguish a speaker. Specially, the last coefficients contain information about the high frequencies that allow capturing the tone of speakers.

Using the cepstral coefficients our method constructs a more concise representation that expresses the speech information by a set of more general and time independent features. In particular, we also characterize the signals by their coefficient's variations. That is, we calculate the change of the coefficient's values between two contiguous signal segments. In order to enrich the acoustic characterization, we also compute the averages of the coefficient's variations as well as their maximum and minimum values. Several experiments were performed with the 16 MFCC and with the 64 different features to represent each signal sample.

Table 1 describes all the 64 statiscally-based features related to each one of the 16 Mel Frequency Cepstral Coefficients. In this table, C_{ik} denotes the coefficient i from segment k, N indicates the number of considered segments, and Δ represents the coefficient variation between contiguous segments.

Table 1. Set of proposed features

Description	Compute	Num. of Features
Maximum value of the coefficient's changes	$\Delta c_i = \max_{k=2}^{N}(c_{ik} - c_{i(k-1)})$	16
Minimum value of the coefficient's changes	$\Delta c_i = \min_{k=2}^{N}(c_{ik} - c_{i(k-1)})$	16
Average value of the coefficient's changes	$\bar{\Delta} c_i = \frac{1}{N-1}\sum_{k=2}^{N} c_{ik} - c_{i(k-1)}$	16
Variance of the coefficient's changes	$\Delta_v c_i = \frac{1}{N-1}\sum_{k=2}^{N}(c_{ik} - \bar{\Delta} c_i)^2$	16

2.2 Statistical Modeling

Once the set of features for every sample of every speaker are computed, we apply a machine learning algorithm to build the identification models. As we mentioned before, we employ two different algorithms: Naïve Bayes and Support Vector Machines. We briefly describe both algorithms.

Naïve Bayes. This probabilistic classifier is based on the assumption that the features are conditionally independent of each other given the target values (classes) [5]. It can be applied to learning tasks where each instance is described by a conjunction of feature values $a_1, a_2, \ldots, a_n$ and the target function f can take any value from some finite set V. That is, given the instance x, the Naïve Bayes classifier assigns it the most probable target value in accordance with the following expression:

$$f(x) = \arg\max_{v_j \in V} P(v_j) \prod_i P(a_i \mid v_j)$$

Where $P(v_j)$ represents the probability that the class v has the *j-th* value and $P(a_i, v_j)$ is the conditionally probability that the feature a takes the *i-th* value given that the class v has the *j-th* value.

Therefore, the construction of a Naïve Bayes classifier basically involves the estimation of the probabilities $P(v_i)$ and $P(a_i \mid v_j)$ from the training data. These estimations are then used to classify new instances using *f(x)*.

Support Vector Machines (SVM). This learning algorithm is specially suited to work with very high dimensional data sets. It uses geometric properties in such a way that it is capable to compute the hyperplane that best separates the training set [6]. In the case where the input space is not lineally separable, it projects the original training space to a higher dimensional feature space using a kernel in order to find an optimal hyperplane. The works by Vapnik [7] and Scholkopf and Smola [8] describe in detail the SVM algorithm.

4 Experiments and Results

In order to prove the proposed method we use a set of the recording samples from the DIMEx100 corpus [9]. This oral corpus for Mexican Spanish contains high quality (44 KHz) recordings from 100 different persons. For each person it includes 50 different phrases of 3.5 seconds long. In total, the corpus is about 291 seconds.

In particular, for the experimental evaluation, we randomly selected 50 persons and constructed speech samples of 3 seconds. In addition, as previously mentioned in section 3, we represented the signals by fixed-size segments of 30ms and characterized each segment using the Mel Frequency Cepstral Coefficients (MFCC).

In order to determine the relevance of the proposed method we performed the following experiments:

Experiment 16MFCC+3. In this case each recording sample was represented by the 16 MFCC coefficients for each segment and three additional features that captures the dynamic behavior of the signal: the minimum, the maximum and the average values for each coefficient.

Experiment 16MFCC+4. This experiment considered the same representation that in the previous case. It only included one additional "dynamic" feature: the variance value of each coefficient.

Experiment 4/16. In this experiment the recording samples were exclusively represented by the set of features expressing the dynamic information of the 16 coefficients, that is, they were represented by the minimum, maximum, average and variance values of each coefficient.

Experiment 4/12. For this final experiment we also only used the features that express the dynamic information, but we only considered 12 MFCC coefficients.

Table 2 shows the obtained results for the four experimental configurations. In all cases we used: (*i*) Naïve Bayes and Support Vector Machines, (*ii*) the Information Gain technique for dimensionality reduction (preserving those features having an information gain greater than zero), and (*iii*) the 10-fold-cross-validation technique for evaluation.

Table 2. Identification accuracies

Experiment	Naïve Bayes	SVM
16MFCC+3	85.59%	85.94%
16MFCC+4	90.47%	89.55%
4/16	94.56%	**97.56%**
4/12	89.59%	94.28%

From table 2 we can derive the following conclusions. First, the dynamic information contained in the Mel Frequency Cepstral Coefficients is very useful for speaker identification. In particular we can observe that using only this information it was possible to achieve an accuracy of 97.56%. Second, using 16 coefficients produced better results than only considering the traditional twelve. This fact indicates that the highest coefficients (the highest frequencies) are relevant for distinguishing among different speakers. Finally, given that both learning algorithms

produced similar results, we presume that the proposed signal characterization is pertinent for the task. In other words, the achieved results were not a direct consequence of the applied classifier.

5 Conclusions and Future Work

This paper described a new method for speaker identification. This method has two main characteristics. On the one hand, it is *text independent* since it does not force users to read a predefined text. On the other hand, the method is *language independent* because it directly works over the speech signal and does not depend on any phonetic segmentation process.

The proposed method is mainly based on a new, simple and compact, signal characterization. This characterization is obtained from the Mel Frequency Cepstral Coefficients of the speech signals and only considers 64 features that resume the *dynamic behavior of the signals*. With this new signal characterization we are able to improve the construction of the speaker's models. Particularly, the presented results indicated an identification accuracy as high as 97%.

Finally, it is important to mention that our results are still preliminary and therefore more experiments are necessary to conclude about the real pertinence of the approach. In order to satisfy this condition we plan to participate in some recognized evaluation forums. Specifically we plan to evaluate our method in the speaker identification task of the NIST forum [10].

Acknowledgments: This work was done under the partial support of CONACYT (project grant 43990). We also thank SNI-Mexico and INAOE for their assistance.

References

1. Reynolds D.A.: An Overview of Automatic Speaker Recognition Technology. In *Proceedings of the IEEE ICASSP, 2002*. Orlando, FL, 2002.
2. García-Perera P., Mex-Perera C. and Nolazco-Flores J. SVM Applied to the Generation of Biometric Speech Key. In *Proceedings of the 9th Iberoamerican Congress on Pattern Recognition (CIARP 2004)*, pages 637-644. Puebla, Mexico, 2004.
3. Campbell W. M., Campbell J. P., Reynolds D. A., Jones D. A. and Leek T. R., Phonetic Speaker Recognition with Support Vector Machines. In *Proceedings of the Neural Information Processing Systems Conference 2003*, pages 1377-1384. Vancouver, BC, Canada, 2003.
4. Bimbot F., Bonastre J-F., Fredouille C., Gravier G., Magrin-Chanolleau I., Meignier S., Merlin T., Ortega-García J., Petrovska-Delacrétaz D. and Reynolds D. A Tutorial on Text-Independent Speaker Verification. In *EURASIP Journal on Applied Signal Processing*, volume 2004, issue 4, pages 430-451, 2004.
5. Mitchell T. *Machine Learning*. McGraw Hill, 1997

6. Stitson M. O., Wetson J. A. E., Gammerman A., Vovk V. and Vapnik V.. 1996. Theory of support vector machines. *Technical Report CSD-TR-96-17*. Royal Holloway University of London, England, 1996.
7. Vapnik V. *The Nature of Statistical Learning Theory*. Number ISBN 0-38794559- Berlin: Springer-Verlag, 1995.
8. Scholkopf B. and Smola A. J. *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press, 2002.
9. Pineda L., Villaseñor-Pineda L., Cuétara J., Castellanos H. and López I. DIMEx100: A New Phonetic and Speech Corpus for Mexican Spanish. In *Proceedings of the IX Ibero-American Conference on Artificial Intelligence (IBERAMIA 2004)*, pages 974-983. Puebla, Mexico, 2004.
10. NIST speaker recognition evaluations. *The NIST year 2006 Speaker Recognition Evaluation Plan*. http://www.nist.gov/speech/tests/spk/index.htm

Agents

Intelligent Virtual Environments for Training: A Tutoring Approach

Raúl A. Aguilar [1], Brunny Troncoso [2], Angélica de Antonio [3] and Ricardo Imbert [3]

[1] Universidad Autónoma de Yucatán,
Mathematics School
Periférico Norte Tablaje 13615, A.P. 172, Cordemex, C.P. 97110,
Mérida, México
avera@uady.mx

[2] Universidad del Bío-Bío,
Information Systems School
Avda. Collao 1202,
Concepción, Chile
btroncos@ubiobio.cl

[3] Universidad Politécnica de Madrid,
Computer Science School, Decoroso Crespo Laboratory
Campus Montegancedo, 28660, Boadilla del Monte,
Madrid, Spain
{angelica,rimbert}@fi.upm.es

Abstract. Scaffolding is proposed as a tutoring approach for Intelligent Virtual Environments for Training. We have used the Activity Theory as a framework to define activities and related tasks to be executed as part of a plan, as proposed in a Team Training Strategy designed by the authors. The strategy is briefly described, and a scaffolding approach is proposed for the Pedagogical Virtual Agent that will integrate the team in the execution stage.

1 Introduction

In the scope of the teamwork, there is a growing understanding of the principles behind effective team training [1]: empirical studies are beginning to tease out the skills that make teams effective (e.g. task skills vs. team skills), the best type of feedback (e.g. outcome vs. process), the best source of feedback (e.g. instructor vs. team mate), the best goal structure (e.g. individual activities vs. group activities). On the other hand, the interaction achieved through software tools has been researched from different perspectives, such as: the learning promoted through human groups attempting to learn something together [2], the activity generated in the shared working space [3], the dialog type shared using intelligent environments [4], and

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 169-179

Received 09/02/07
Accepted 08/04/07
Final version 20/04/07

others. For Barros in [5], technology is interesting as far as it has potential to create, to favor, or to enrich interpersonal contexts of learning.

A Collaborative Virtual Environment (CVE) is a computer-based, distributed, virtual space or set of places; in a CVE people can meet and interact with others, with agents or with virtual objects [6]. CVEs —specially, Intelligent Virtual Environments for Training (IVETs)— can be used to train one or more students in the execution of a certain task, particularly in situations in which training in the real environments is either impossible or undesirable because it is costly or dangerous. The intelligence in the IVETs usually falls on a component named Pedagogical Virtual Agent (PVA) [7].

In the Decoroso Crespo Laboratory, in line with own interest on computer science applied to education and training, we have proposed a Team Training Strategy (TTS) —briefly described in section 2— which promotes collaborative learning skills as well as knowledge and skills in a procedurally oriented domain. The learning goals are organized in tasks, using the Activity Theory (AT) —briefly introduced in section 3— as a framework. The TTS will be conducted with the assistance of an IVET in each one of the stages. Regarding the team training principles above mentioned, in this approach: the type of feedback is centered in the process; the team mate (PVA) is in charge of the task skills tutoring; and we use group activities —cooperative activities— for organizing the goal structure

In this paper, we will focus on the third stage of the TTS. The section 4 describes the activities and the IVET proposed for assisting the team during the execution stage. In the section 5, the tutoring approach proposed for the PVA is described in detail. Finally, the conclusions and ongoing work of this proposal are presented in the last section.

2 The Team Training Strategy

The strategy consists of five interrelated stages in which the human team to be trained follows an iterative process of self-assessment about the execution of a plan to perform a proposed task (see fig. 1).

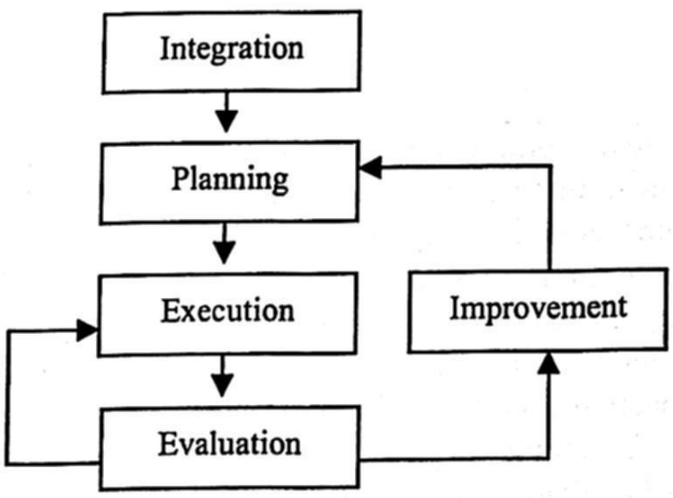


Fig. 1. The Team Training Strategy (TTS)

The *Integration* stage has the purpose of integrating the human team (using a CVE), as well as providing the apprentices (assisted by a Human Tutor) with a first mental schema of the plan to be executed for a predefined task. In the following stage (*Planning*) the team members, in a collaborative virtual meeting, co-construct an execution plan for the task (using a planning tool). In the third stage (*Execution*), the team uses an IVET to execute the planned activities according to their assigned roles; in this environment a PVA plays a team leader role to assist the trainees. In the fourth stage (*Evaluation*), the team members will have to evaluate their previous execution and must identify both individual and group errors with the purpose of avoiding them in a future execution stage. Finally, in the last stage (*Improvement*) the team members, in a virtual meeting, co-construct a new plan for the task using as a baseline the experience acquired during the iterative execution and evaluation of the initial plan.

The level of members interaction is expected to keep on growing while the team makes progress through the strategy stages (see fig. 2) and the iterative process followed by the team will trigger mechanisms that will generate a shared mental model between its members, enhancing the collaborative learning and allowing a better group performance during teamwork.

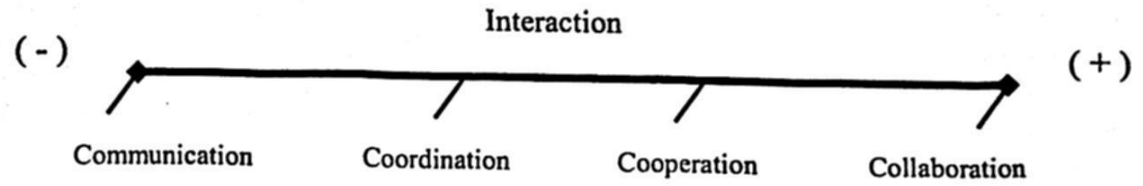


Fig. 2. Degree of Interaction

The potential domains for the application of the TTS are the control and maintenance of industrial or nuclear plants, operation and maintenance of industrial equipment, coordinated manipulation of vehicles and, especially, human rescue teams for disaster situations. The strategy orients the trainee to acquire simultaneosly: knowledge, skills, and attitudes, in the three domains of learning promoted by training (cognitive, psychomotor, and relational/social).

3 Activity Theory

Jonassen & Rohrer-Murphy, in [8], consider that the AT has its roots in the classical German philosophy of Kant and Hegel, which emphasized both the historical development of ideas and the active and constructive role of humans.

AT states that all human activities are mediated by culturally created signs or tools; through external interactions with these signs the inner mental state of the person is transformed, the knowledge is interiorized [9]. Kuutti in [10] explains that the activity is the basic unit of the model proposed by Leont'ev, he presents a hierarchy of three levels in which the activity is at the upper level, the actions at the middle level, and the operations at the lower level.

Initially, the development of the AT tried to explain individual activities, however, most of the human activities are collective. Thus, Engeström [11] extends the model to consider collective activities; according to this new model, an activity is decomposed into one or more tasks (actions), has an objective, and is developed by a subject (individual or group) using a tool (physical or abstract tool) according to the rules of the community in which the activity is performed.

AT provides a powerful framework for studying and understanding human activities. In a CVE for training, AT can be used to think of how to support the training activities being transformed into automated operations, and to think of how the work objective is divided among the different learners and how to best support the associated actions. We propose in this paper an "activity set" to define the different activities available in the CVE for the trainee; each element belonging to the activity set allows the access to a related "action set", which has all the actions regarding to an activity, each one properly represented by a Learning Object.

4 The IVET for the Execution Stage in the TTS

At the *Execution* stage, the group members should achieve the previously described goal, executing the planned activities according to their assigned roles. To perform the task, the team will use an IVET which recreates a scenario in which the apprentices must carry out sequential or concurrent activities, according to the established plan.

In order to assist the team during this stage, we propose to include a PVA playing the role of team leader. It may communicate with the apprentices or make them suggestions during the execution of their activities, if necessary. The PVA will offer its help to the team, giving preference to critical activities for the task success. Fig. 3 shows a team member's view in an IVET prototype during a training session.

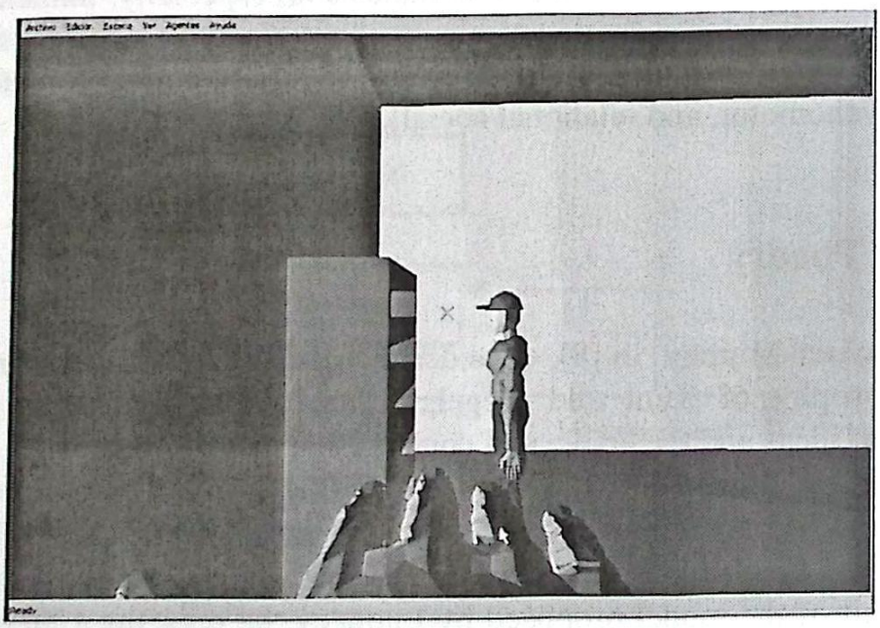


Fig. 3. A trainee view in an IVET prototype

Our development is based on MAEVIF [12], a Software Architecture aimed to develop IVETs. This architecture has been devised to be open and flexible, and basically it's composed by two subsystems: one dealing with the graphical visualization of the virtual environments and the interaction with the learners; and the other being a multi-agent system designed to provide "intelligence" to the tutoring system. For the development of the specific IVET proposed for the execution phase, the MAEVIF architecture is being adjusted extending the multi-agent system with a mechanism to model the group (Group Modeling Agent), inexistent in the original structure of MAEVIF.

Furthermore, the incorporation of a PVA as a team member requires to model it appropriately in order to provide it with behaviors similar to the ones expected in a human partner or in a human tutor. With this aim we have chosen to design and build our PVA according to a multilayered agent architecture named COGNITIVA [13], which allows the PVA to produce and manage reactive, deliberative and social behaviors influenced by personal characteristics and emotions.

5 The Scaffolding Approach for PVAs in IVETs

A scaffold is a temporary structure that physically supports workers while they complete jobs that would otherwise be impossible; when the work has finished, the scaffold is removed. The instructional scaffolding metaphor is used to describe a teaching strategy and some kinds of supports offered to the trainees in their interaction with teachers, tutors and partners when they are in a learning activity, focusing on the effective intervention of an "advanced partner".

According to Vygotsky [14], the zone in which the student can solve problems with external help (because s/he is ready to capture new contents) is called Zone of Proximal Development (ZPD), and is formally defined like the distance between the actual development level (determined by the student grade of independent problem solving) and the possible development level (determined by the student grade of problem solving with advanced help).

The scaffolding metaphor is attractive for IVETs because it focuses the attention at the trainer role in the apprenticeship process, and makes emphasis in each treinee individually. Scaffolding can be applied as a tutoring approach for PVAs in IVETs according to the steps described in the following subsections.

5.1 Activity Selection

In order to apply scaffolding successfully in the IVET it is necessary to classify the activities and actions. Firstly, we can consider the activity set which contains the different activities available in the IVET for the trainees (1); secondly, each element belonging to the activity set allows access to a related "action set", containing all the actions possibly related to an activity, each one properly represented by a Reusable Learning Object (RLO) (2).

$$\text{Activity set} = \{Act_1, Act_2, \ldots, Act_n\}, \text{ where } n > 0 \quad (1)$$

$$\text{Action set}_i = \{RLO_{i1}, RLO_{i2}, \ldots RLO_{ij}\}, \quad (2)$$

where i, j >0 and "i" refers to the activity "i"

The trainee's activity set will contain a limited number of activities according to what the trainee is expected to do in his/her role. We will use the syntax proposed in the Design by Contract Method [15], sketched as follows:

```
rlo <URI>
        require
                precondition 1
                ...
        ensure
                postcondition 1
                ...
```

5.2 Construction of the SKM and ZPD

When the trainee starts his/her training activity the PVA will have to define correctly the Zone of Proximal Development (ZPD) and the Student Knowledge Model (SKM) associated to him/her. We define a RLO repository in the system which can be accessed by the PVA. All RLOs are accessible to the PVA by their associated metadata. The best procedure to solve a problem, the SKM and the ZPD will be constructed using these associated RLO metadata. The PVA uses the repository to build the trainee's SKM and to suggest activities to him. The SKM is defined as a set of components, where each component makes reference to a RLO completed by the trainee (3).

$$SKM = \left\{RLO_1^k, RLO_2^k, \ldots, RLO_l^k\right\}, where\ l \geq 0 \quad (3)$$

The PVA constructs the trainee's ZPD based on the SKM; the process starts by looking in the RLO repository for the RLOs that can be satisfied in all (or almost all) of their preconditions with the post-conditions of the RLOs that belong to the SKM. Then we can construct the trainee's ZPD as a RLO set, where each element will reference a specific RLO the trainee is ready to learn with partner's help (4). Finally, the PVA is able to know what the trainee is ready to learn chooses the proper activities, and suggest the trainee to practice those which he/she needs to reinforce.

$$ZDP = \left\{RLO_1^D, RLO_2^D, \ldots, RLO_p^D\right\}, where\ p \geq 0 \quad (4)$$

5.3 Action Demonstration

Once one activity has been selected, the PVA can execute a complete action demonstration, giving explanations and repeating it any times the trainee needs. The PVA shows the trainee how to perform the activity in a step-by-step way, using available resources like the verbalization of the problem solving plan. On the other hand, the scaffolding proposed fully solves the problem just at the beginning of the training, and then offers partial solutions to the problem or gives cues when the trainee needs it.

A way to implement this approach in an IVET is allowing the full activity demonstration at the beginning, but limiting afterwards the demonstration to the action currently performed.

5.4 Skill Imitation and Action Assessment

This step provides the training experience to the trainee; here s/he has the opportunity to execute the actions by him/herself. All the time the trainee is executing the activity, the PVA is assessing his/her performance, ready to give scaffoldings if these are needed (repeated failures, too long time without actions, etc.). The amount and types of scaffoldings offered could change depending on the trainee's performance; for example, the PVA can: provide advices, explanations or suggestions inviting him/her to do something; give her/him cues or ideas, etc. The PVA can remove the scaffoldings allowing the trainee to perform the action by him/herself, whenever it considers it appropriate.

An activity is finished when the trainee can execute it correctly without any help; this can occur at any moment during the training and the IVE gives each trainee the possibility to train any times s/he needs.

5.5 Trainee ZDP and SKM Updating

The PVA must apply a continuous assessment over the learner's actions and it's in charge of updating continuously the ZPD and SKM of the trainee with their successes or failures. Using these previously defined structures, the PVA can check which actions the trainee could execute (or complete) comparing the RLOs belonging to the ZPD to the RLOs associated with each defined action. In the same way, comparing the SKM to each action set, the PVA is able to know which actions have been learnt and determine which activities are complete, which ones are incomplete, and which ones have not been trained at all. To suggest an action to execute, the PVA will prefer the ones related to the activity in course; anyway, if the trainee decides to change the activity or the action, the PVA has the mechanism to offer adequate alternatives.

5.6 End of the Training

The trainee will be invited to review a summary about his/her work session when s/he decides to end the training (s/he can review previous sessions if s/he wants to). Presenting the trainee an overview about their training performance is a powerful way to meet the goal of having the trainee in charge of their own progress; in this way s/he can engage her/himself in an active training perspective, processing the information in a deeper level of understanding and being able to recognise what s/he does not know yet.

5.7 Applying the Model

Virtual Reality Technology is especially valuable in domains where real life training is impossible, very expensive or where students can experience some risky situations [16], such a maintenance or control of Nuclear Power Plants (NPP). Méndez et al. have described in [17] two applications for training in NPPs, MAEVIF and PRVIR, which help the trainee to learn how to perform physical, procedural task, such as the procedure for entrance in a radioactivity controlled area.

Even if the trainee has freedom to walk around the environment and select the activity and the actions to do, each action has a very well defined number of steps to be followed, with a clear sequence of RLOs. Let's consider, for example, some activities, actions and roles defined in a NPP which allow us to illustrate the proposed structures:

NPP Activities:
(access_NPP, access_radioactivity_area, operate_machine1, ...)

Related Actions:

Access_NPP_Actions=(identification_process, access_individual_box,

put_on_working_suit, put_on_security_shoes)

Access_radioactivity_area_Actions = (put_on_security_gloves,

put_on_security_mask, hold_tool)

...

Related Roles:
Role<learner-operator>

Require

Lrn.type=operator

Ensure

Lrn.knows(Access_NPP)

Lrn.knows(access_radioactivity_area)

```
        Lrn.knows(operate_machine1)
        ...

Role<learner-maintenance>
Require
        Lrn.type=maintenance
        Lrn.knows(Access_NPP)
        Lrn.knows(access_radioactivity_area)
        Lrn.knows(operate_machine1)
        ...
Ensure
        Lrn.knows(machine_maintenance)
Role<learner-supervisor>
Require
        Lrn.type=supervisor
        Lrn.knows=machine_maintence
Ensure
        Lrn.knows=performance_supervision
```

Each action set is composed of several actions that the trainee has to perform to meet the activity goal. Each action in the IVET is described by one RLO, which describes the sequence and the tools used to correctly execute the action in the plant. The RLOs are stored in a repository; each RLO can be accessed through the contract associated metadata. Not forgetting that there are pre-requisites and post-conditions for each RLO, the Access_radioactivity_area_Actions have as pre-requisite the well-executed Access_NPP_Actions; in the same way, having correctly executed the Access_NPP_Actions gives the trainee the faculty of walking around the power plant installations.

As the trainee is working in the IVET, the PVA is conveying his successes and failures to the related agents, in order to update the individual associated Binnacle, SKM_set and ZPD_set. According to the previous example, let suppose the following trainee performance, recorded in its associated binnacle (see table 1).

As a consequence of the trainee's performance, some actualization of SKM_set and ZPD_set is done:

```
SKM_set = (Access_NPP)
ZPD_set = (Access_radioactivity_area, ...)
```

All over the trainee's learning activities, the PVA is tutoring the learner according to both the current strategy step and the trainee's performance and experience. These

are important factors to determine the kind of responses the PVA will offer (quantitative and qualitatively).

Table 1. The Trainee binnacle.

Trainee_ID	Date	RLOi-executed	Start-time	End-time	Success
xx1	2005-10-10	identification_process	14:45	15:05	True
xx1	2005-10-10	access_individual_box	15:20	15:35	False
xx1	2005-10-10	access_individual_box	15:35	15:45	True
xx1	2005-10-10	wear_working_suit	15:50	16:05	True
xx1	2005-10-10	wear_security_shoes	16:10	16:35	True
...					

6 Conclusions and Ongoing work

In this paper, Scaffolding as a Tutoring Approach for a PVA has been proposed in a TTS. The TTS has been designed taking as a framework the AT and considering instructional design aspects. Using the AT too, we have also proposed some useful structures, like the Student Knowledge Model, the Zone of Proximal Development and the activities and actions sets. Then, we have showed the advantages of using these structures in an IVE to formulate a Scaffolding Tutoring proposal for a PVA. The PVA have characteristics that allow increasing the computer's ability to engage and motivate trainees along their training process [18].

Once we have formulated and designed these structures, our current work is focused on the implementation of the CVE for Training (CVET) based on the TTS and its related Scaffolding Tutoring Approach, in order to check the viability/effectiveness of our proposal. As a first step of this development, we have already designed a software architecture for IVEs allowing the execution of the training practice using a virtual 3D environment. This virtual 3D environment will allow to carrying new experimentation with the TTS considering the lessons learnt in [19], and the proposed tutoring approach.

References

1. Cannon-Bowers, J. & Salas, E.: Making Decisions Under Stress. Implications for individual and team training. Wash DC: APA. (1998)
2. Dillenbourg, P.: What do you mean by collaborative learning?. In P. Dillenbourg (Ed) Collaborative-learning: Cognitive and Computational Approaches. Oxford: Elsevier (1999)

3. Mühlenbrock, M.: Action-based collaboration analysys for group learning. Dissertations in AI Program, University of Duisburg, The Netherlands, IOS Press. (2001)
4. Soller, A.: Supporting Social Interaction in an Intelligent Collaborative Learning System. International Journal of Artificial Intelligence in Education, 12(1). (2001) 40-62
5. Barros, B.: Aprendizaje colaborativo en enseñanza a distancia: Entorno genérico para configurar, realizar y analizar actividades en grupo. PhD. Thesis. Universidad Politécnica de Madrid. (1999)
6. Snowdon, D., Churchill, E.F. & Munro, A.J.: Collaborative Virtual Environments: Digital Spaces, Places for CSCW: An Introduction. In Churchil, E.F, Snowdon, D. & Munro, A.J. (eds), Collaborative Virtual Environments: Digital Places and Spaces for Interaction. London: Springer -Verlag. (2001) Chapter 1.
7. Rickel, J. & Johnson, W.: Animated Agents for Procedural Training in Virtual Reality: Perception, Cognition, and Motor Control. Applied AI. 13. (1999) 343–382
8. Jonassen, D. & Rohrer-Murphy, L: Activity theory as a framework for designing constructivist learning environment. Educational Technology, Research and Development, 47 (1), Research Library. (1999) 61-79
9. Fjeld, M., Lauche, K., Bichsel, M, Voorhorst, F., Krueger, H., Rauterberg, M.: Physical and Virtual Tools: Activity Theory Applied to the Design of Groupware, in Computer Supported Cooperative Work. Vol.11. (2002) 153-180
10. Kuutti, K.: The concept of activity as a basic unit of analysis for CSCW research, Proceedings of the second European Conference on Computer Supported Cooperative Work. Amsterdam, The Netherlands. (1991) 249-264.
11. Engeström, Y.: Learning by Expanding: An Activity-Theoretical Approach to Developmental Reseach. Helsinky, Finland: Orienta Konsultit Oy. (1987)
12. De Antonio, A., Ramírez, J., Imbert, R., Méndez, G. & Aguilar, R.: A Software Architecture for Intelligent Virtual Environments Applied to Education. Revista de la Facultad de Ingeniería, Univ. de Tarapacá. Vol. 13 No. 1, Arica: Chile. (2005) 47-55
13. Imbert, R. & De Antonio A.: When Emotion Does Not Mean Loss of Control. IVA'2005. 5th International Working Conference on Intelligent Virtual Agents. T. Panayiotopoulos, J. Gratch, R. Aylett, D. Ballin, P. Olivier, T. Rist (eds.). Lecture Notes in Artificial Intelligence, vol. 3661, Springer. Kos, Grecia. (2005) 152-165
14. Vygotsky, L. S. Mind in society: The development of higher psychological processes. Cambridge, MA: Harvard University Press. (1978)
15. Sánchez-Alonso, S. & Sicilia, M.A.: Relationships and commitments in learning object metadata. Proceedings of the 5th International Conference on Information Technology Based Higher Education and Training: ITHET 2004. Istambul, Turkey. (2004)
16. Pantelidis, V.S.: Suggesstions on when to use and when not to use virtual reality in education. VR in the schools, 2 (1), (1996) 18
17. Méndez, G., Herrero, P. and de Antonio, A.: Intelligent Virtual Environments for Training in Nuclear Power Plants. Proceedings of the 6th International Conference on Enterprise Information Systems, Portugal. (2004)
18. Aguilar, R.A., de Antonio, A. & Imbert, R..: Pedagogical Virtual Agents to Support Training of Human Groups. Proceedings of Electronics, Robotics, and Automotive Mechanics Conference (CERMA 2006). Cuernavaca, México. (2006) Vol. 1. 149–154
19. Aguilar, R.A., de Antonio, A. & Prieto, M.: A procedure for the evaluation of instructional techniques used for the integration of teams. Proceedings of IADIS International Conference (CELDA 2005). Oporto, Portugal. (2005) 459–462

Computing, Interaction and Multi-Agent Systems: A Formal Language Approach

Gemma Bel-Enguix and M. Dolores Jiménez-López

Research Group on Mathematical Linguistics (GRLMC)
Rovira i Virgili University
Pl. Imperial Tárraco, 1, 43005 Tarragona, Spain
gemma.bel@urv.cat
mariadolores.jimenez@urv.cat

Abstract. Starting from the metaphor of computation as *interaction*, we introduce a multi-agent system framework based on a non-classical formal language model and inspired in one of the most common forms of interaction: *dialogue*.

1 Introduction

According to [5], computing has enjoyed several different metaphors for the notion of computation. Until the mid-1960s, most people thought of computation as *calculation*, or operations undertaken on numbers. From the 1960s, computation was re-conceptualised more generally as *information processing*, or operations on text, audio or video data. With the growth of the Internet over the last fifteen years, a new metaphor for computation has appeared: computation as *interaction*.

In order to exploit this new metaphor of computing, many people deal with agent technologies. Agent-based systems are one of the most important areas of research and development that have emerged in information technology in the 1990s. Roughly speaking, an agent is a computer system that is capable of flexible autonomous action in dynamic, unpredictable, multi-agent domains.

The concept of agent can be found in a range of disciplines as, for example, computer networks, software engineering, artificial intelligence, human-computer interaction, distributed and concurrent systems, mobile systems, telematics, information retrieval, etc. In general, multi-agent systems offer strong models for representing complex and dynamic real-world environment.

According to [5], agent technologies can be grouped into three categories, according to the scale at which they apply:

1. *Organization-Level*: technologies and techniques related to agent societies as a whole. Here, issues of organizational structure, trust, norms and obligations and self-organization in open agent societies are paramount.
2. *Interaction-Level*: technologies and techniques that concern the communications between agents –for example, technologies related to communication languages, interaction protocols and resource allocation mechanisms.

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 181-190

Received 09/02/07
Accepted 08/04/07
Final version 19/04/07

3. *Agent-Level*: technologies and techniques concerned only with individual agents –for example, procedures for agent reasoning and learning.

In this paper we introduce a multi-agent system model based on grammar systems that can be placed at the *interaction-level*. In general, problems solved by technologies on this level have been studied in other disciplines such as economics, political science, philosophy and linguistics. In fact, the framework we present in this paper has been inspired on linguistics, specifically on the functioning of natural language dialogue. Natural Language can be seen not just a system of communication but also as a coordination system. The capacity of natural language to play these two roles has been used in order to develop an interactive model that integrate coordination and communication in a multi-agent system.

Throughout the paper, we assume that the reader is familiar with the basics of formal language theory, for more information see [8] and [7].

2 Conversational Grammar Systems

Grammar systems theory is a consolidated and active branch in the field of formal languages [1] that provides syntactic models for describing multi-agent systems at the symbolic level, using tools from formal grammars and languages. The attempt of the 'parents' of the theory was, as themselves state in [1], *'to demonstrate a particular possibility of studying complex systems in a purely syntactic level'* or, what is the same, to propose a *grammatical* framework for multi-agent systems. Grammar systems theory has been widely investigated and nowadays constitutes a well-developed formal theory that presents several advantages with respect to classical models. However, being a branch of formal languages, researchers in the field of grammar systems have concentrated mainly on theoretical aspects. Roughly speaking, a grammar system is a *set* of grammars working together, according to a specified protocol, to generate a language. Notice that while in classical formal language theory *one* grammar (or automata) works individually to generate (or recognize) *one* language; here, instead, we have *several* grammars working together in order to produce *one* language.

While grammar systems are related to AI, a subfield of the theory, –the so-called eco-grammar systems– is closely related to Artificial Life. Eco-grammar systems provide a syntactical framework for eco-systems, this is, for communities of evolving agents and their interrelated environment. Briefly, an eco-grammar system is defined as a multi-agent system where different components, apart from interacting among themselves, interact with a special component called 'environment' [6].

Here we introduce a new model: *Conversational Grammar Systems* (CGS). CGS are multi-agent systems based on grammar systems, specifically in the so-called eco-grammar systems. Conversational grammar system offer a framework with a high degree of flexibility, what means that they are able to accept new concepts and modify rules, protocols and settings during the computation.

Evolution and action are involved in a consistent way in environment/contexts, while interaction of agents with the medium is constant. Moreover, conversational grammar systems present the following advantages to exploit the idea of computation as interaction:

- generation process is highly *modularised* by a distributed system of contributing agents;
- it is *contextualized*, linguistic agents re-define their capabilities according to context conditions given by mappings;
- and *emergent*, it emerges from current competence of the collection of active agents.

In what follows we introduce the formal definition of our model.

3 Elementary Components

In conversational grammar systems we distinguish two types of components: *agents* and *environment*. These elementary components are defined in the following way:

Definition 1 *A Conversational Grammar System (CGS) of degree n, $n \geq 2$, is an $(n+1)$-tuple:*

$$\Sigma = (E, A_1, ..., A_n),$$

where:

- $E = (V_E, P_E)$,
  - V_E *is an alphabet;*
  - P_E *is a finite set of rewriting rules over* V_E.
- $A_i = (V_i, P_i, R_i, \varphi_i, \psi_i, \pi_i, \rho_i)$, $1 \leq i \leq n$,
  - V_i *is an alphabet;*
  - P_i *is a finite set of rewriting rules over* V_i;
  - R_i *is a finite set of rewriting rules over* V_E;
  - φ_i: $V_E^* \rightarrow 2^{P_i}$;
  - ψ_i: $V_E^* \times V_i^+ \rightarrow 2^{R_i}$;
  - π_i *is the start condition;*
  - ρ_i *is the stop condition;*
  - π_i *and* ρ_i *are predicates on* V_E^*. *We can define the following special types of predicates. We say that predicate* σ *on* V_E^* *is of:*
    - *Type* (a) *iff* $\sigma(w) = true$ *for all* $w \in V_E^*$;
    - *Type* (rc) *iff there are two subsets* R *and* Q *of* V_E *and* $\sigma(w) = true$ *iff* w *contains all letters of* R *and* w *contains no letter of* Q;
    - *Type* (K) *iff there are two words* x *and* x' *over* V_E *and* $\sigma(w) = true$ *iff* x *is a subword of* w *and* x' *is not a subword of* w;

* *Type* (K') *iff there are two finite subsets* R *and* Q *of* V_E^* *and* $\sigma(w) =$ *true iff all words of* R *are subwords of* w *and no word of* Q *is a subword of* w;
* *Type* (C) *iff there is a regular set* R *over* V_E *and* $\sigma(w) =$ *true iff* $w \in R$.

The items of the above definition have been interpreted as follows: a) E represents the environment described at any moment of time by a string w_E, over alphabet V_E, called the *state of the environment.* The state of the environment is changed both by its own evolution rules P_E and by the actions of the agents of the system, A_i, $1 \leq i \leq n$. b) A_i, $1 \leq i \leq n$, represents an agent. It is identified at any moment by a string of symbols w_i, over alphabet V_i, which represents its current state. This state can be changed by applying evolution rules from P_i, which are selected according to mapping φ_i and depend on the state of the environment. A_i can modify the state of the environment by applying some of its action rules from R_i, which are selected by mapping ψ_i and depend both on the state of the environment and on the state of the agent itself. Start/Stop conditions of A_i are determined by π_i and ρ_i, respectively. A_i starts/stops its actions if context matches π_i and ρ_i. Start/stop conditions of A_i can be of different types: (a) states that an agent can start/stop at any moment. (rc) means that it can start/stop only if some letters are present/absent in the current sentential form. And (K), (K') and (C) denote such cases where global context conditions have to be satisfied by the current sentential form.

4 Elementary Configurations

In CGS, we define an elementary configuration as a *state* in which the system can be at a given time.

Definition 2 *A state of a CGS* $\Sigma = (E, A_1, \ldots, A_n)$, $n \geq 2$, *is an* $n+1$*-tuple:*

$$\sigma = (w_E; w_1, \ldots, w_n),$$

where $w_E \in V_E^*$ *is the state of the environment, and* $w_i \in V_i^*$, $1 \leq i \leq n$, *is the state of agent* A_i.

5 Behavior of Agents

The behavior of agents in CGS consists in the application of action rules to the state of the environment. We describe this behavior as a sequence of *context-change-actions* allowed by the current environment and performed by two or more *agents.* An *action* is defined as the application of a rule *on the environmental string.* This rule is applied to the state of the environment *by an active agent,* and it is not any rule, but a rule selected by $\psi_i(w_E, w_i)$, that is, a rule (an action) allowed by the current context and by the state of the agent itself. We define an *active agent* in relation to the allowable actions it has at a given moment. Formally:

Definition 3 *By an action of an active agent A_i in state $\sigma = (w_E; w_1, w_2, \ldots, w_n)$ we mean a direct derivation step performed on the environmental state w_E by the current action rule set $\psi_i(w_E, w_i)$ of A_i.*

Definition 4 *An agent A_i is said to be active in state $\sigma = (w_E; w_1, w_2, \ldots, w_n)$ if the set of its current action rules, that is, $\psi_i(w_E, w_i)$, is a nonempty set.*

6 Environmental Dynamics

Since interaction in CGS is understood in terms of *context changes*, we have to define how the environment passes from one state to another as a result of agents' actions:

Definition 5 *Let $\sigma = (w_E; w_1, \ldots, w_n)$ and $\sigma' = (w'_E; w'_1, \ldots, w'_n)$ be two states of a CGS $\Sigma = (E, A_1, \ldots, A_n)$. We say that σ' arises from σ by a simultaneous action of active agents $A_{i_1}, \ldots, A_{i_r}$, where $\{i_1, \ldots, i_r\} \subseteq \{1, \ldots, n\}$, $i_j \neq i_k$, for $j \neq k$, $1 \leq j, k \leq r$, onto the state of the environment w_E, denoted by $\sigma \overset{a}{\Longrightarrow}_\Sigma \sigma'$, iff:*

- *$w_E = x_1 x_2 \ldots x_r$ and $w'_E = y_1 y_2 \ldots y_r$, where x_j directly derives y_j by using current rule set $\psi_i(w_E, w_{i_j})$ of agent A_{i_j}, $1 \leq j \leq r$;*
- *there is a derivation:*

$$w_E = w_0 \overset{a}{\Longrightarrow}^*_{A_{i_1}} w_1 \overset{a}{\Longrightarrow}^*_{A_{i_2}} w_2 \overset{a}{\Longrightarrow}^*_{A_{i_3}} \cdots \overset{a}{\Longrightarrow}^*_{A_{i_r}} w_r = w'_E$$

such that, for $1 \leq j \leq r$, $\pi_{i_j}(w_{j-1}) = true$ and $\rho_{i_j}(w_j) = true$. And for $f \in \{t, \leq k, \geq k\}$ the derivation is:

$$w_E = w_0 \overset{a}{\Longrightarrow}^f_{A_{i_1}} w_1 \overset{a}{\Longrightarrow}^f_{A_{i_2}} w_2 \overset{a}{\Longrightarrow}^f_{A_{i_3}} \cdots \overset{a}{\Longrightarrow}^f_{A_{i_r}} w_r = w'_E$$

such that, for $1 \leq j \leq r$, $\pi_{i_j}(w_{j-1}) = true$[1], and

- *$w'_i = w_i$, $1 \leq i \leq n$.*

However, in the course of the computation, agents' states are also modified and the environmental string is subject to changes due to reasons different from agents' actions. So, in order to complete our formalization, we add the following definition:

Definition 6 *Let $\sigma = (w_E; w_1, \ldots, w_n)$ and $\sigma' = (w'_E; w'_1, \ldots, w'_n)$ be two states of a CGS $\Sigma = (E, A_1, \ldots, A_n)$. We say that σ' arises from σ by an evolution step, denoted by $\sigma \overset{e}{\Longrightarrow}_\Sigma \sigma'$, iff the following conditions hold:*

- *w'_E can be directly derived from w_E by applying rewriting rule set P_E;*

[1] In this latter case the stop condition $\rho_i(w_j) = \text{true}$ is replaced by the stop condition given the f-mode.

- w_i' *can be directly derived from* w_i *by applying rewriting rule set* $\varphi_i(w_E)$, $1 \leq i \leq n$.

In CGS, computation implies that both the *state of the environment* and *state of agents* change. Such changes take place thanks to two different types of processes: *action steps* and *evolution steps*. By means of the former, active agents perform actions on the environmental string modifying its state; the latter imply the reaction of context and agents which, according to the changes produced by agents' actions, modify their states. So, action steps and evolution steps alternate in the course of the computation. At the end, what we have is a *sequence of states* reachable from the initial state by performing, alternatively, action and evolution derivation steps:

Definition 7 *Let* $\Sigma = (E, A_1, \ldots, A_n)$ *be a CGS and let* σ_0 *be a state of* Σ. *By a state sequence (a derivation) starting from an initial state* σ_0 *of* Σ *we mean a sequence of states* $\{\sigma_i\}_{i=0}^{\infty}$, *where:*

- $\sigma_i \overset{a}{\Longrightarrow}_\Sigma \sigma_{i+1}$, *for* $i = 2j$, $j \geq 0$; *and*
- $\sigma_i \overset{e}{\Longrightarrow}_\Sigma \sigma_{i+1}$, *for* $i = 2j + 1$, $j \geq 0$.

Definition 8 *For a given CGS* Σ *and an initial state* σ_0 *of* Σ, *we denote the set of state sequences of* Σ *starting from* σ_0 *by* $Seq(\Sigma, \sigma_0)$.

The set of environmental state sequences is:

$$Seq_E(\Sigma, \sigma_0) = \{\{w_{Ei}\}_{i=1}^{\infty} \mid \{\sigma_i\}_{i=0}^{\infty} \in Seq(\Sigma, \sigma_0), \sigma_i = (w_{Ei}; w_{1i}, \ldots, w_{ni})\}.$$

The set of state sequences of the j*-th agent is defined by:*

$$Seq_j(\Sigma, \sigma_0) = \{\{w_{ji}\}_{i=1}^{\infty} \mid \{\sigma_i\}_{i=0}^{\infty} \in Seq(\Sigma, \sigma_0), \sigma_i = (w_{Ei}; w_{1i}, \ldots, w_{ji}, \ldots, w_{ni})\}.$$

$Seq(\Sigma, \sigma_0)$ describes the behavior of the system, this is, the possible state sequences, directly following each other, starting from the initial state. $Seq_E(\Sigma, \sigma_0)$ and $Seq_j(\Sigma, \sigma_0)$ are the corresponding sets of sequences of the states of the environment and of the states of j-th agent, respectively.

Now, we associate certain languages with an initial configuration:

Definition 9 *For a given CGS* Σ *and an initial state* σ_0 *of* Σ, *the language of the environment is:*

$$L_E(\Sigma, \sigma_0) = \{w_E \in V_E^* \mid \{\sigma_i\}_{i=0}^{\infty} \in Seq(\Sigma, \sigma_0), \sigma_i = (w_E; w_1, \ldots, w_n)\}.$$

and the language of j*-th agent is:*

$$L_j(\Sigma, \sigma_0) = \{w_j \in V_A^* \mid \{\sigma_i\}_{i=0}^{\infty} \in Seq(\Sigma, \sigma_0), \sigma_i = (w_E; w_1, \ldots, w_j, \ldots, w_n)\}.$$

for $j = 1, 2, \ldots, n$.

$L_E(\Sigma, \sigma_0)$ and $L_j(\Sigma, \sigma_0)$ correspond to those states of the environment and to those states of the j-th agent, respectively, that are reachable from the initial configuration of the system.

7 Interaction Protocol

Coordination is defined in many ways but in its simplest form it refers to ensuring that the actions of independent actors (agents) in an environment are coherent in some way. The challenge therefore is to identify mechanisms that allow agents to coordinate their actions. Research to date has identified a huge range of different types of coordination and cooperation mechanisms, raging from emergent cooperation, coordination protocols to distributed planning. In CGS, we define different *modes of derivation* that can be seen as the interaction protocol of our multi-agent system:

Definition 10 *Let $\Sigma = (E, A_1, ..., A_n)$ be a CGS. And let $w_E = x_1x_2...x_r$ and $w'_E = y_1y_2...y_r$ be two states of the environment. Let us consider that w'_E directly derives from w_E by action of active agent A_i, $1 \leq i \leq n$, as shown in Definition 5. We write that:*

$w_E \overset{a}{\Longrightarrow}^{\leq k}_{A_i} w'_E$ *iff* $w_E \overset{a}{\Longrightarrow}^{\leq k'}_{A_i} w'_E$, *for some* $k' \leq k$;
$w_E \overset{a}{\Longrightarrow}^{\geq k}_{A_i} w'_E$ *iff* $w_E \overset{a}{\Longrightarrow}^{\leq k'}_{A_i} w'_E$, *for some* $k' \geq k$;
$w_E \overset{a}{\Longrightarrow}^{*}_{A_i} w'_E$ *iff* $w_E \overset{a}{\Longrightarrow}^{k}_{A_i} w'_E$, *for some* k;
$w_E \overset{a}{\Longrightarrow}^{t}_{A_i} w'_E$ *iff* $w_E \overset{a}{\Longrightarrow}^{*}_{A_i} w'_E$ *and there is no* $z \neq y$ *with* $y \overset{a}{\Longrightarrow}^{*}_{A_i} z$.

In words, $\leq k$-derivation mode represents a time limitation where A_i can perform at most k successive actions on the environmental string. $\geq k$-derivation mode refers to the situation in which A_i has to perform at least k actions whenever it participates in the derivation process. With $*$-mode, we refer to such situations in which agent A_i performs as many actions as it wants to. And finally, t-derivation mode represents such cases in which A_i has to act on the environmental string as long as it can.

One way of getting transitions with no gap and no overlap in CGS is to endow agents with an *internal control* that contains start/stop conditions that allow agents to recognize places where they can start their activity, as well as places where they should stop their actions and give others the chance to act. This is, start/stop conditions help agents to recognize *transition relevance places*, i.e. places where speaker change occurs. Start/stop conditions have been formally defined in Definition 1.

Figure 1 gives a graphic idea of the multi-agent system architecture we have introduced.

8 Example

The following simple example illustrates how CGS work.

Example 1 *Consider the following CGS: $\Sigma = (E, A_1, A_2)$, where:*

- $E = (V_E, P_E)$,

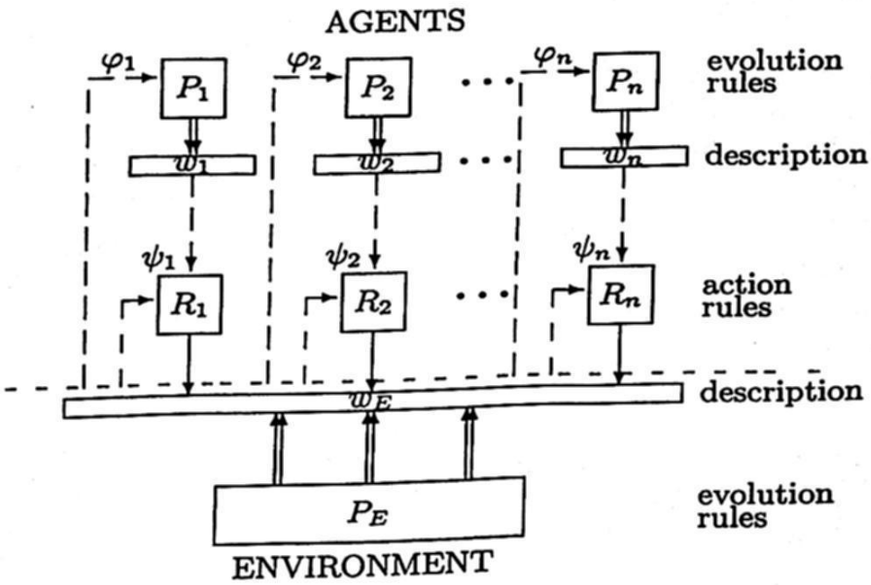


Fig. 1. Conversational Grammar Systems.

- $V_E = \{a, x, y\}$;
- $P_E = \{a \to b^2, b \to a^2, x \to x, y \to y\}$.

– $A_1 = (V_1, P_1, R_1, \varphi_1, \psi_1, \pi_1, \rho_1)$ *with:*
- $V_1 = \{c\}$;
- $P_1 = \{c \to c\}$; $R_1 = \{a \to x\}$;
- $\varphi_1(w) = P_1$ *for every* $w \in V_E^*$;
- $\psi_1(w; u) = R_1$ *for* $w \in \{a, x, y\}^*$ *and* $u = c$, *otherwise* $\psi_1(w; u) = \emptyset$;
- $\pi_1 =$ *true for all* $w \in V_E^*$; $\rho_1 =$ *true for all* $w \in V_E^*$.

– $A_2 = (V_2, P_2, R_2, \varphi_2, \psi_2, \pi_2, \rho_2)$ *with:*
- $V_2 = \{d\}$;
- $P_2 = \{d \to d\}$; $R_2 = \{b \to y\}$;
- $\varphi_2(w) = P_2$ *for every* $w \in V_E^*$;
- $\psi_2(w; v) = R_2$ *for* $w \in \{b, x, y\}^*$ *and* $v = d$, *otherwise* $\psi_2(w; v) = \emptyset$;
- $\pi_2 =$ *true for all* $w \in V_E^*$; $\rho_2 =$ *true for all* $w \in V_E^*$.

P_E, P_1 and P_2 contain rules of an 0L system applied in a parallel way. Rules in R_1 and R_2 are pure context-free productions applied sequentially. Let us suppose that the system is working in the arbitrary mode $*$. And let us take $\sigma_0 = (a^3; c, d)$ as the initial state of Σ. Then, a possible derivation in Σ is the following one:

$$(a^3; c, d) \overset{a}{\Longrightarrow}^*_\Sigma (a^2x; c, d) \overset{e}{\Longrightarrow}^*_\Sigma (b^4x; c, d) \overset{a}{\Longrightarrow}^*_\Sigma (yb^3x; c, d) \overset{e}{\Longrightarrow}^*_\Sigma$$
$$(ya^6x; c, d) \overset{a}{\Longrightarrow}^*_\Sigma (ya^2xa^3x; c, d) \overset{e}{\Longrightarrow}^*_\Sigma \cdots$$

Notice, that we alternate action and evolution steps. At every action step one of the agents rewrites one symbol of the environmental state, while in evolution steps both environmental and agents' states are rewritten according to 0L rules.

9 Final Remarks

Grammar Systems provide a well-defined *theoretical* formal model for multi-agent systems with interesting and well-known formal results. Taking into account this formal language model and considering the functioning of interaction in natural language we have defined a new model called *conversational grammar systems*. The core ideas of the model we have introduced here are the following ones:

- *Multi-agent System.* Conversational grammar systems can be seen as a multi-agent system for computation. CGSs define systems of distributed components in which components can be viewed as autonomous problem solvers that must collaborate in order to perform complex tasks.
- *Distribution.* We have a multi-agent system of *autonomous* agents where the functionality of an agent is viewed as an emergent property of its intensive interaction with its dynamic environment. Each autonomous agent may accomplish its own task, or cooperate with other agents, to perform its own individual task or a global social one.
- *Dynamic, emergent.* Conversational grammar systems offers a dynamic and emergent model for interaction. Conversational grammar systems have been defined as a set of agents developing their activity on a common shared environment. And we have emphasized very much the fact that actions performed by agents are determined both by the state of context and by the state of the agent itself. We have not postulated any external control to fix the sequence of actions to be performed during derivation process. Which actions must be performed at any moment is a matter solved *locally* and *opportunistically,* by taking into account which the state of context at that precisely moment is.

We claim that CGS provides a powerful framework for formalizing any kind of *interaction*, both among agents and among agents and the environment. A topic where context and interaction among agents is essential is the field of dialogue modelling and its applications to the design of effective and user-friendly computer dialogue systems where we think our model can be directly applied.

Our model has been defined as a multi-agent system. Agent technology is one of the fastest growing areas of information technology. People agree on the fact that the apparatus of agent technology provides a powerful a useful set of structures and processes for designing and building complex software applications. The metaphor of autonomous problem solving entities cooperating and coordinating to achieve their objectives is a natural way of conceptualizing many problems. Being conversational grammar systems an agent-based model, they share all those advantageous features. Therefore, we think that our model may contribute to the field of agent technologies by offering a highly formalized framework that could be applied to many different issues.

References

1. Csuhaj-Varjú, E., Dassow, J., Kelemen, J. & Păun, Gh. (1994), *Grammar Systems: A Grammatical Approach to Distribution and Cooperation*, Gordon and Breach, London.
2. Helander, M.G., Landauer, T.K. & Prabhu, P.V. (1997), *Handbook of Human-Computer Interaction*, Elsevier, Amsterdam.
3. Jennings, N.R. & Wooldridge, M.J. (1998), *Agent Technology. Foundations, Applications and Markets*, Springer, Berlin.
4. Kraus, S. (1997), Negotiation and Cooperation in Multi-Agent Environments, *Artificial Intelligence*, 94, 79-97.
5. Luck, M., McBurney, P., Shehory, O. & Willmott, S. (eds.) (2005), *Agent technology: Computing as Interaction. A Roadmap for Agent Based Computing*, University of Southampton (AgentLink III).
6. Păun, Gh. (ed.) (1995), *Artificial Life: Grammatical Models*, Black Sea University Press, Bucharest.
7. Rozenberg, G. & Salomaa, A. (1997), *Handbook of Formal Languages*. Springer, Berlin.
8. Salomaa, A. (1973), *Formal Languages*, Academic Press, New York.
9. Tatai, G. & Gulyás, L. (eds.) (1999), Agents Everywhere, Springer, Budapest.

Multi-Agent Based Control for Urban Traffic Networks

Joel A. Trejo Sánchez and Julio C. Ramírez Pacheco

Universidad del Caribe, SM 78, Manzana 1, Lote 1, 77528, Cancún QRoo, México
{jtrejo,jramirez}@ucaribe.edu.mx

Abstract. This paper deals with the traffic management within an urban traffic network (UTN). The paper shows the necessity of an adaptive control system that detects and takes decisions when traffic congestion occurs. A multi-agent control scheme is proposed in which a control agent is charged to manage the traffic lights of an intersection; the agent communicates with other agents of the neighbor intersections. The architecture of such a control agent is described as well as relevant issues of a prototype implementation. The simulation test of an adaptive control policy is presented.

Keywords: Urban traffic control; Agent based approach.

1 Introduction

Urban traffic congestion causes considerable cost due to time losses, has a negative impact on the environment and increases the probability of accidents [1]. Coupled with an expanding population, urban traffic networks are unable to efficiently handle the daily movements of traffic through urban areas. Adjusting the traffic lights according with the traffic conditions is a costly solution if it involves human interaction. Traffic light management is a complex problem and there exist several intelligent algorithms that have been used to solve it.

The main urban traffic control objectives are to maximize the performance of the system and to reduce the total time of accumulated delay through real-time decision making processes. A way to reach these objectives is the updating of traffic lights timing according to the demand of traffic and an appropriate coordination strategies among the local traffic controllers within a specific area.

In this paper a multi agent based approach for addressing the problem of traffic control is presented. An agent is charged to control the operation of a traffic light in an intersection; it determines the timing and policies according to the vehicle flow in the streets and the current functioning of other traffic light controller agents in the traffic network.

This paper is organized as follows. Section 2 overviews the background of the presented work. Section 3 describes the control system integration with an urban traffic simulator. Section 4 presents the architecture for traffic control. Some experimental results using micro simulation are presented in section 5.

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 191-201

Received 23/02/07
Accepted 08/04/07
Final version 20/04/07

2 Urban Traffic Control

Adaptive control systems must have the capability to adjust the traffic signals based on current traffic. There exist several approaches to implement an adaptive traffic control system; some of them are based on learning refoircement, fuzzy logic, multi-agent systems(MAS), discrete events systems, and others approaches. The proposed model is based on multi-agent systems because of the agent's capacities including autonomy, interaction, and modularity of the software.

In [2] a Petri net model is proposed to control the traffic lights. The model is divided in two categories: 1) to determine the signal sequence to follow in the control policy and 2) how to implement the logic to manage the signals. Models based on hybrid Petri nets are proposed in [3] and in [4]. Traffic flow and density are represented with continuous Petri nets, and traffic lights are modeled as discrete event models.

Three types of agents for the traffic management are proposed by [5]. The intersection agents decide the pertinent control strategy of the intersections based on their objectives, capacities, perception, and data. In [6] a multi-agent methodology is developed focusing on learning reinforcement to distribute the traffic control. With the aim of coordinate the intersections, in [7] is proposed the use of mobile agents to set coordination with the rest of the intersections.

3 An Urban Traffic Simulator

In order to prove our methodology, a way to measure our results is needed. Nowadays analysis and optimization of traffic systems are preferred as an alternative to develop a traffic control system; analytical and simulation methods have been proposed for the study of such systems since it permits to predict the control system behavior in the real world.

There exist two main simulation approaches: event oriented and time (slicing) oriented. Our traffic control methodology is integrated into the discrete event approach since it allows more flexibility and advantages over the time oriented [8].

3.1 Interaction With the Simulator

The simulator was developed in Java using the facilities of *ProActive* (a Java GRID middleware library for parallel, distributed and multi-threaded computing) for communication and managing the connection among various nodes (modules). When the simulation engine is executed it registers in a port of the local machine, allowing registering other node through the network if it is included in the grid.

As we have mentioned above, the simulator can be modeled as a discrete event system. The way in which the simulator works is as follows: each time an event occurs in the traffic network, this event models a change in the state of the system and is responsible to generate more events which depend on it. Each event has a time in which it is expected to be triggered. When events occur, they represent actions to be

executed on the system. The events are stored in a future event list (FEL). Next picture shows graphically the previous ideas.

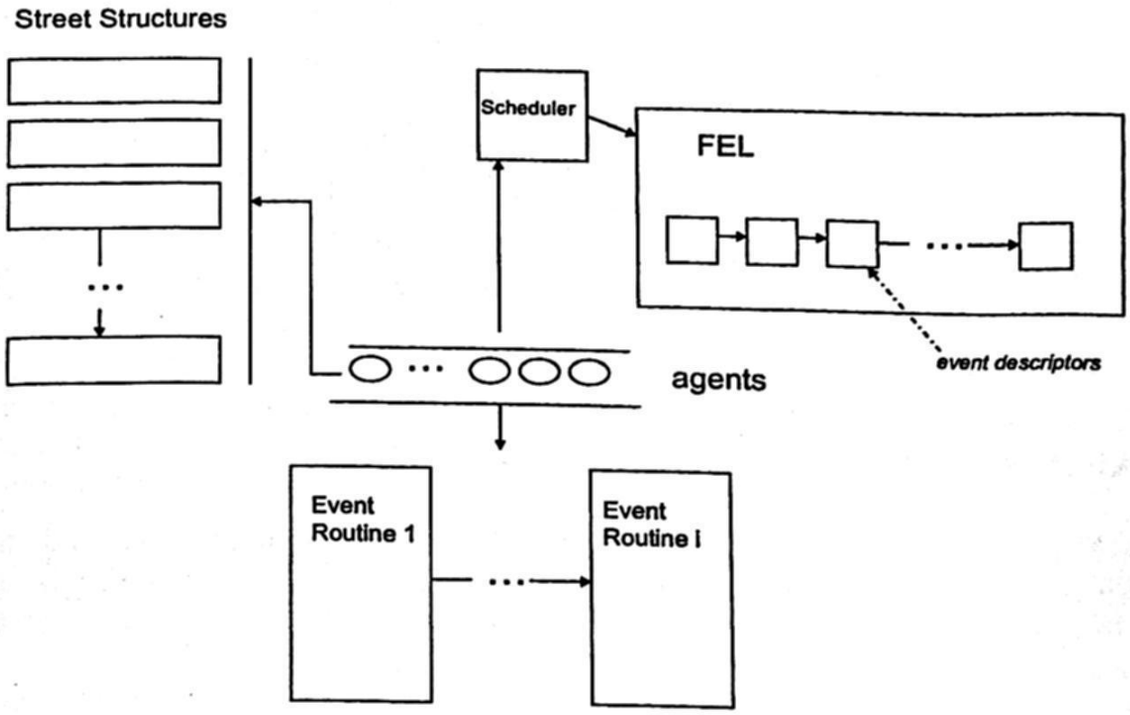


Fig. 1 Simulation Engine

The simulation engine generates a file configuration to save the results of the simulation (executed activities), including the following information: *Vehicle Identifier, Event Time, Link, Node, Lane, Distance, Velocity, Event Type, Acceleration, Driver, Passengers*, etc. that allows to get performance indexes such as traffic density, flow, etc.

Each time an activity is executed in the simulation engine this module sends the pertinent information about such activity to all the modules connected to the middleware. This resumes the operation of both control and visualization modules. The visualization module uses this information to process and visualize the events in a graphical manner. The control module uses this information to take decisions according to the state of the streets.

The control module is composed by several control agents. The control agent interacts with its environment, and it is capable to interact with others agents when it is necessary to coordinate a control policy. In figure 2 the interaction between the simulator and the control module is illustrated. This interaction is performed through a middleware that handles a future event list (FEL). The system operates as follows: the control module generates a set of events representing the light change instants, according to the current control policy; these events are stored in the FEL; the interface provides to the control module the information on traffic flow and density yield during the simulation execution through the middleware. This allows the agent the updating of the phase time of the current control policy.

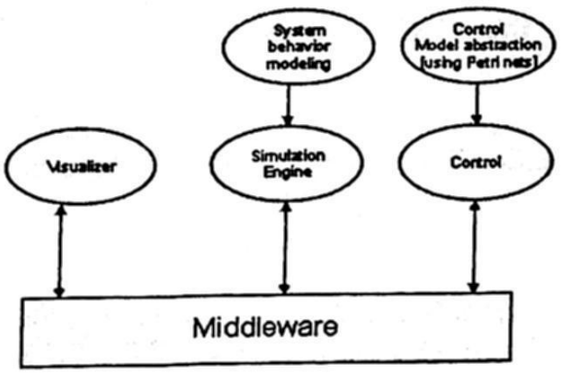


Fig. 2 Interaction between the control module and simulator

4 An Agent Architecture for Traffic Control

The urban traffic network approach is composed by several intersections and streets; each intersection is controlled by an agent with the following features: *interaction with its environment, decision making, proactive, communication with other agents in the system, unable to solve the problem by itself.*

The control agent is composed by several interacting modules charged to perform specific functions. In figure 3 it is shown the control agent architecture. The input of the control agent is provided by the simulation engine; this information regards the current traffic flow and density of all the streets in the traffic network. The output consists of an event sequence regarding the traffic light change instants; this sequence is added to the FEL.

Every control agent is concerned with the involved streets in the intersection where the agent is operating. Now the paper describes the functionality of each one of the agent's components.

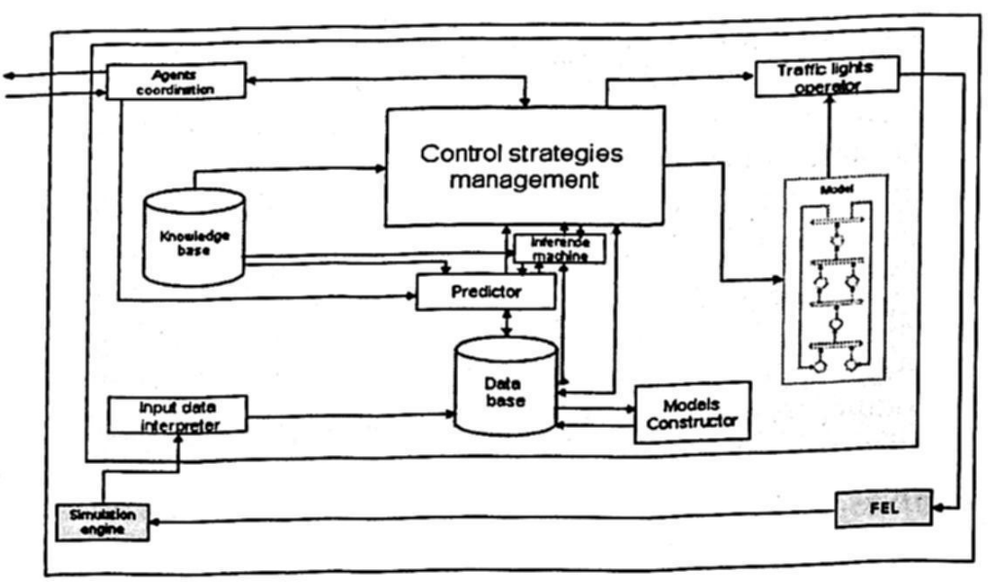


Fig. 3 Control Agent Architecture

4.1 The Control Agent Components

- The data base includes the initial configuration of the intersections, namely, the intersection layout, the allowed phases (stages of a control policy), initial time assigned to phases, and the state of the streets related to intersection.
- The model constructor is charged to transform (offline) the control policies stored on data base into timed Petri net models, to be used by other components of the agent.
- The input data interpreter analyzes the input information and computes traffic flow and density values on the adjacent streets to the intersection that is controlled by the agent.
- The knowledge base includes the rules decision used by the control management module and predictor modules to take decisions about the controller behavior. The rules decisions are organized as subsets according to the kind of decisions. Access to each group of rules is determined by metarules.
- The predictor computes future traffic conditions according to rules decision and supported by the surveillance function. The surveillance function ensures that the predictions are precise.
- The control strategies management module proposes the control policy according to the observed traffic conditions on the intersection. This module receives as input the flow Φ_i and density δ_i for every street *i* involved on intersection; the input is analyzed and then the module can decide if the current control policy must be changed.
- The model describes the current control policy defining the operation of the traffic light in the intersection; this policy is determined by the control management module. The model is a timed Petri net that specifies which movements have right of way in each phase of the policy, and the time elapse assigned to each phase.
- The traffic light operator executes the policy model. The sequence of events (firing of transitions) and their instants are translated into relevant events for the simulator; they are added to the FEL respecting the order of the events already included in the list.
- The agent coordination module is responsible to establish the communication with the other agents informing about the current control policy or requesting the performance indexes of streets concerning other intersections.

4.2 Control Agent Operation

The controller is a proactive agent that operates according the current situation of the environment simulation. It applies a pertinent control policy and delivers to the simulation engine the traffic lights behavior in the form of an event list. The behavior of the control agent is determined by the control strategies management module, which coordinates the rest of the agent components.

The agent adaptation to new traffic conditions is performed by the change of the control policies. This change may be accomplished in two ways: a) updating the timing of a current policy or b) replacing the control policy by another one.

Timing Updating

The policies management module decides, according to the traffic demand in the involved streets, if the current time elapses assigned to the phases of the policy must be modified. The change consists in the updating of some or all delays associated to places in the Petri net model of the policy.

Policies Replacement

When the change of traffic conditions is significant, the control management module may decide to substitute the current policy with another one, which is best adapted to the traffic behavior; the module requests to the model execution module to stop the current policy, updates the Petri net model with the new policy, and resumes the activity of the execution module.

Combining Policies

It is possible to define complex policies including more than one phase sequence and alternate their operation triggered by external events representing namely, pedestrian request, different traffic conditions, and priority vehicles. In figure 4 the model shown includes two sequences of phases (which can be considered as two policies).

The execution module may fire the sequence $\sigma 1$= (t1 t2 t3 t4 t5 t6)* while the external event associated to tc1 does not appear; when such an event is present then tc1 fires and the sequence $\sigma 1$=(T1' T2' T3' T4')* is fired. The back to execute the first sequence of phases may be accomplished when tc2 (according to the associated external event) is fired.

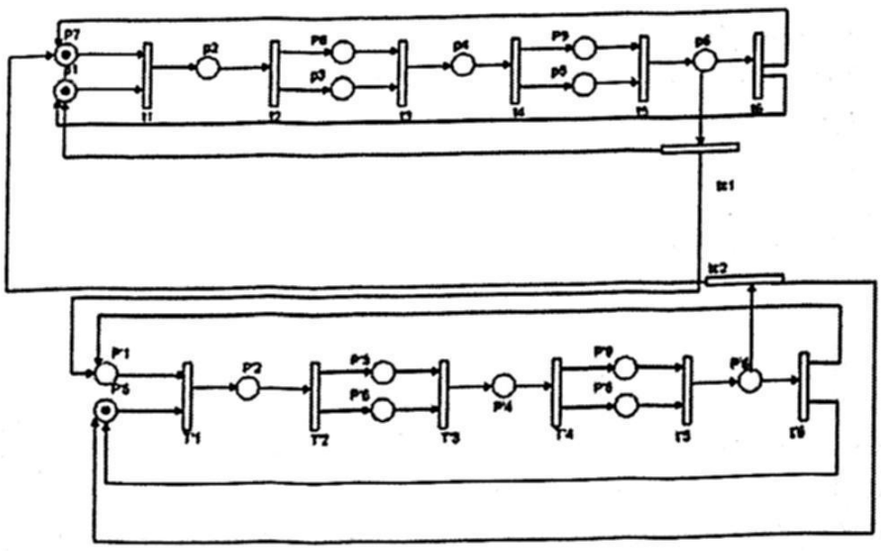


Fig. 4 Timed Petri Net model including two sequences of phases

Agent Interaction

Any change in the control policies, as described above, must be informed to the neighbor control agents through the interaction module. This information may be used by these agents to initiate a policy adaptation process. This means that if an agent perceives saturation on any street belonging to the intersection, this agent can decide to interact with other agents on adjacent intersections to decide the change of a control strategy.

5 Implementation and Results

The traffic control agent has been developed according to the proposed architecture using Java. A control strategy is executed by the traffic lights operator according to the state equation for the timed Petri net model. Traffic lights operator translates the transition triggers into significant light change events. The interaction with the rest of the simulator is performed through active objects from the middleware. Some methods are defined for communication between agents: *lookup_agent, start_dialog,accept_invitation, reject_invitation and send_information.*

Several tests have been performed on a LAN in order to test the functioning of the control agents. For illustration purposes the paper includes the simulation results obtained in a simple case study.

Consider the scheme of figure 5; it describes four intersections separated by a distance of 1Km. Parameters referring to the vehicles speed, random distributions of the vehicles, and others, are modified to observe the behavior of the control agent, when the traffic conditions change.

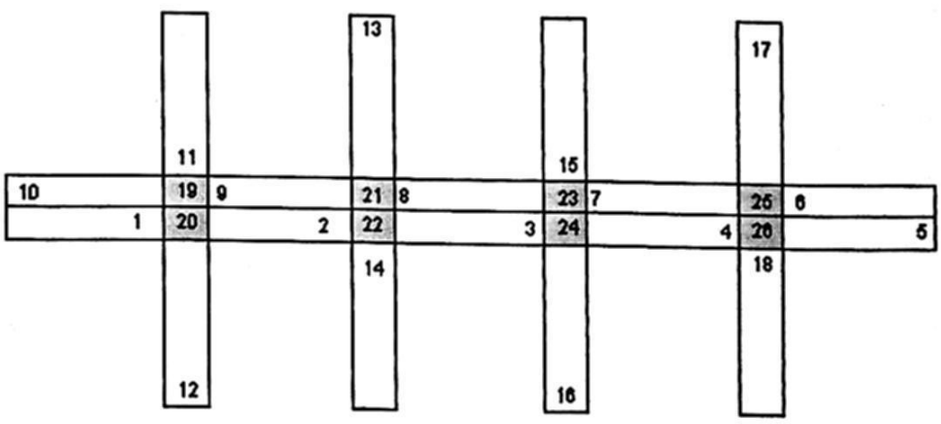


Fig 5 Traffic Network

Figures 6 and 7 show respectively, the starting and the ending of a simulator execution for the network described above. When simulation starts the queue length in the inflow streets is higher than outflow streets as we can see in figure 6; but at the end of the simulation, we can notice that the queue length on all streets is similar (figure 7). This suggests that the traffic conditions get stable.

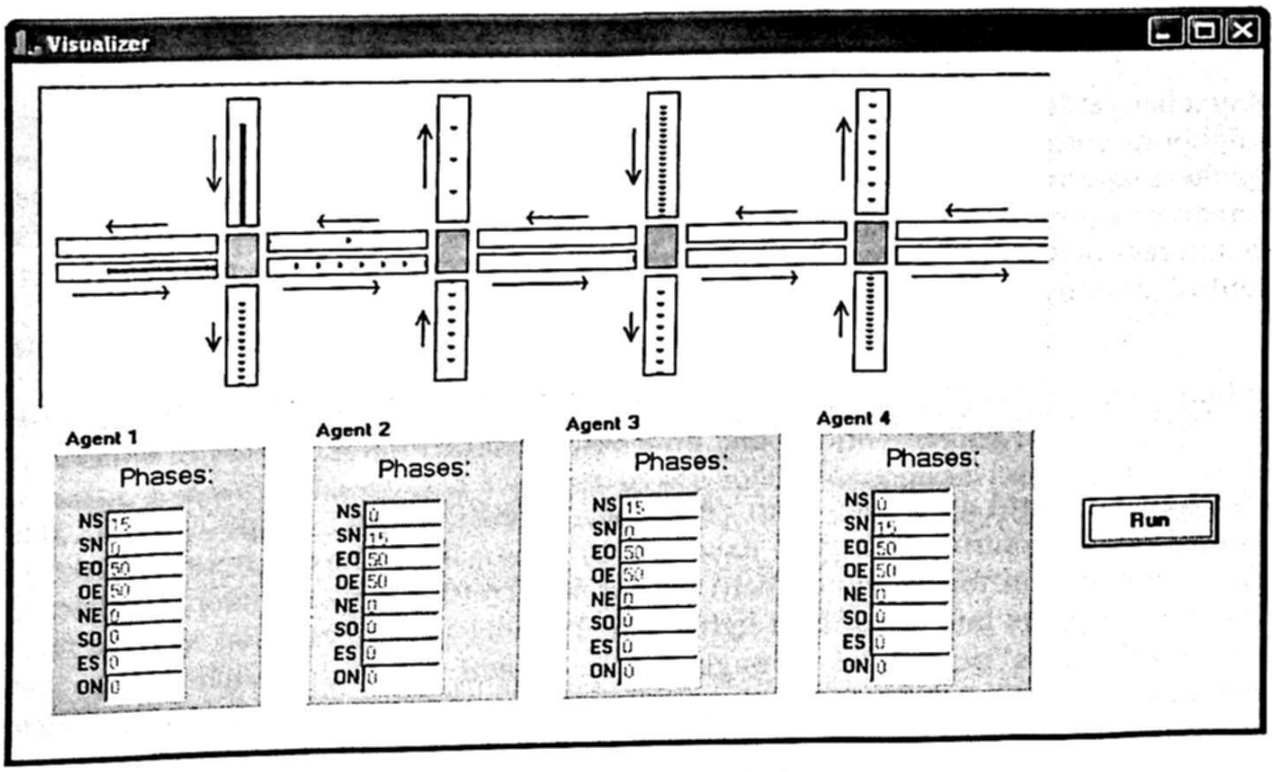


Fig. 6 Start of the simulation

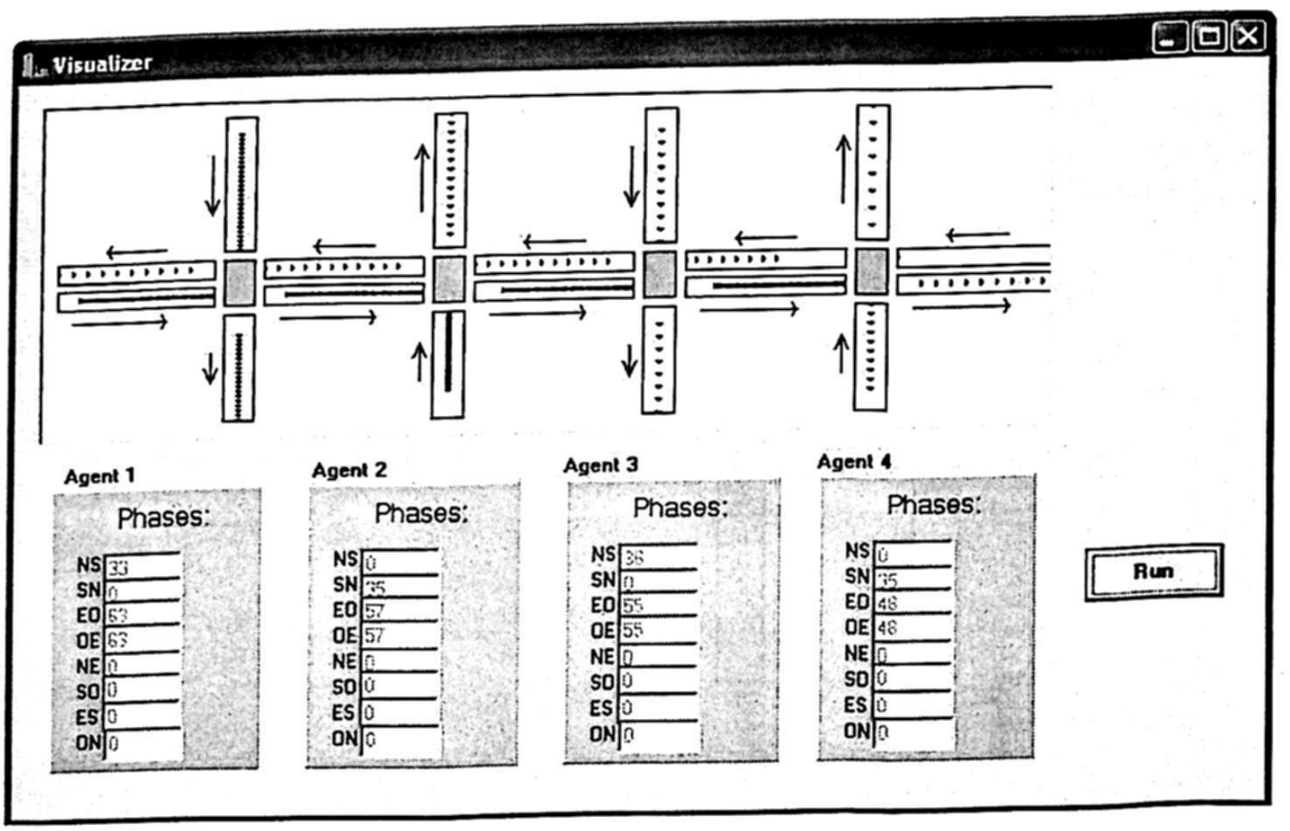


Fig. 7 End of simulation

In figure 8 the graphic shows the flow and density of vehicles regarding the 1 and 11 street segment, using a fixed control policy. The two-phase policy assigns 50 seconds to the west-east phase (link 1) and 15 seconds to the north-south phase (link 2) during

all the simulation time. We can notice in both graphics the flow and density values are variable during the simulation time; that indicates that the fixed-time policy it is not appropriate for variable traffic conditions.

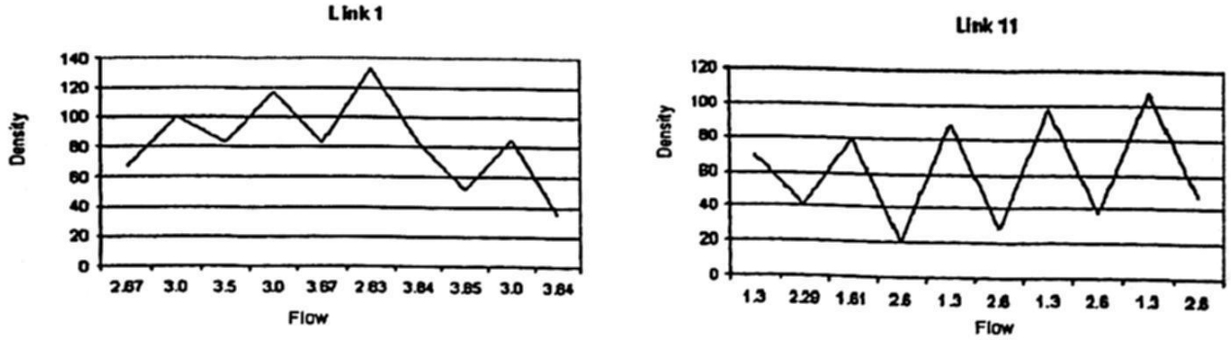


Fig. 8 Pretimed control for street segment 1 and street segment 11

Now we are going to analyze the results of applying an adaptive time policy. In figure 9 it is shown the flow and density of vehicles for the 1 and 11 street segments respectively; vertical lines in graphic represents the instant in which the times assigned to phases are updated. In the graphics we can notice several stages: at the beginning, density is high and flow is low; then while simulation time is growing the density is reduced and trends to be constant and the flow is increased and also trends to be constant during the rest of simulation; that is flow and density values trend to stabilize because of the dynamic updating of the time elapses assigned to phases.

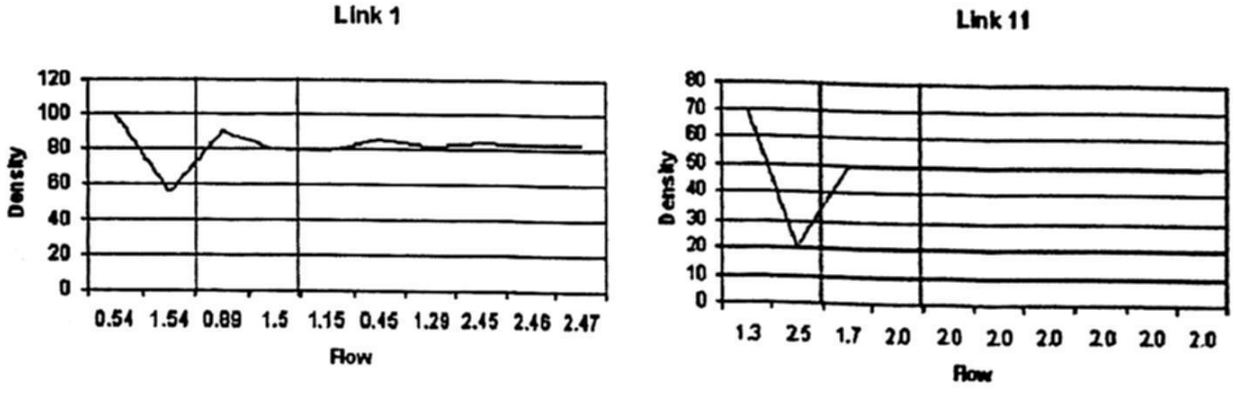


Fig. 9 Time updating control for street segment 1 and street segment 11

Now the paper presents the results of applying an adaptive control, in which besides of the times updating control, the model could be replaced by a model more appropriated to the traffic conditions. In figure 10 it is shown the flow and density of vehicles for the 1 and 11 street segments respectively; the first vertical lines in each graphic represents that the model has been replaced. As we can notice in the graphics, after the control policies are replaced, traffic flow increase and density decrease, but density and flow values are variable; when the time assigned to phases is updated then the density and flow values trend to be constant. As we can notice, the traffic conditions trends to stabilize when an adaptive control strategy is established.

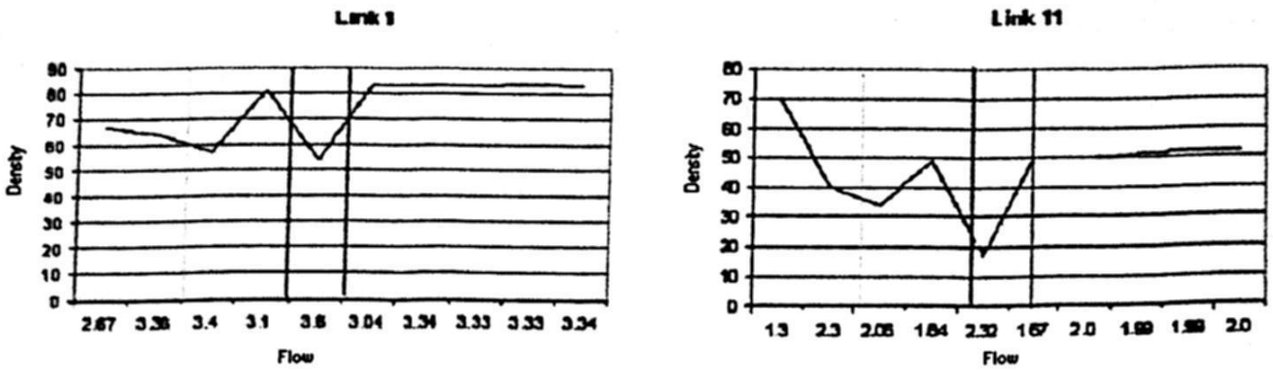


Fig. 10 Policies control replacement for street segment 1 and street segment 11

6 Conclusions

Urban traffic control is proposed based on a multi-agent approach. In our scheme the traffic light of each intersection is controlled by an agent; this agent proposes a control strategy according to the traffic demand. The control agent interacts with other agents assigned to the neighbor intersections. As we noticed during several tests the performance indexes of a traffic network are vastly improved when dynamic strategies for updating control policies were adopted.

Current research address: a) complex control policies dealing with external events (pedestrian requests, priority vehicles), and b) the definition of coordination strategies among control agents.

References

1. Carbone C. ."Control Oriented Discrete Event Models for Manufacturing and Transport Systems". Phd Thesis, Facoltà di Ingegneria Elettrica. 2003-2004.
2. G.F. Cetin, M. "Modeling traffic signal control using Petri nets". Intelligent Transportation Systems, IEEE Transactions on, Sept 2004, pp 177-187.
3. Julvez J., Boel R.. "Modelling and Controlling Traffic Behaviour with Continuous Petri Nets". D.G.A. ref B106/2001 and a European Community Marie Curie Fellowship, CTS, contract number: HPMT-CT-2001-00278.
4. DiFebbraro Angela, Giglio D, Sacco. "Urban Traffic Control Structure Based on Hybrid Petri Nets". Intelligent Transportation Systems, IEEE transactions on, Dec 2004, pp 224-237.
5. Danko A. Roozemond , Jan L.H. Rogier. "Agent controlled traffic lights". ESIT 2000, 14-15 September 2000, Aachen, Germany.
6. Marco Wiering, Jelle Van Veenen, Jilles Vreeken, Arne Koopman. "Intelligent Traffic Light Control". Institute of information and computing sciences, Utrecht University Technical Report. UU-CS-2004-029
7. Fei-Yue Wang. "Agent-Based Control for Networked Traffic Management Systems". IEEE Computer Society September 2005. Vol 20, No. 5. pp 92-96.

8. Schulze T., Fliess T., "Urban Traffic Simulation with psycho-physical vehicle-following models", 1997, En Proceedings of the 1997 Winter Simulation Conference ed. S. Andradottir, K. J. Healy, D. H Withers, and B. L. Nelson.
9. E. López-Neri, E. López-Mellado, A. Ramírez-Treviño. "Hierarchical Models of Urban Traffic Systems for Agent Based Simulation". International Workshop Optimization of Urban traffic Systems. Julio 2005, pp.1-12
10. Nwana H. S., Software Agents: An Overview, In Knowledge Engineering Review, Cambridge University Press, 1996.
11. Proactive V3.0.1 Documentation.
12. Fujimoto, R. M. 1999. Parallel and distributed simulation. In Proceedings of the 31st Conference on Winter Simulation: Simulation---A Bridge To the Future - Volume 1 (Phoenix, Arizona, United States, December 05 - 08, 1999).p-p 122-131.
13. Wooldridge M. "Intelligent Agents ". In G.Weiss, Editor: Multiagent Systems (MIT Press 1999).

Time Series

On a New Approach to Time Series Tracking

Karinne Ramírez-Amaro and Juan Carlos Chimal-Eguía

Centro de Investigación en Computación, Av. Juan de Dios Bátiz s/n Unidad Profesional Adolfo López Mateos Col. Nueva Industrial Vallejo, Mexico City, 07738, Mexico,
kramireza@ipn.mx, chimal@cic.ipn.mx

Abstract. In this paper we present a new approach to time series tracking using some main ideas from the paper published by Donald Michie in 1963. One of those ideas stands that is easier to solve a complex problem by dividing it into many easy subproblems that are sequentially linked. Using this idea we divide the time series in small parts, this divisions will provide information of the amplitude behavior in order to "track" the time series. The obtained results indicate that using this new approach it is possible to track the time series with high accuracy and also deduce more information about the time series. This approach could be applied to any natural phenomena which can be represented as time series.

1 Introduction

In 1963, Donald Michie published a paper that describes a trial and error machine which learns to play the game "Noughts and Crosses" [1]. This device was initially constructed from matchboxes and colored beads as shown in Fig. 1. This machine is a perfect example of the game theory application. The game theory is an interesting topic since the games provided a microcosm of intellectual activity. Those thought processes which we regard as being specifically human accomplishments, such as learn from experience, inductive reasoning, argument for analogy, the formation and testing of hypotheses, are brought to into play even in the simple games of mental skill.

As an example of this, the matchbox machine was used for a particular mental activity of trial and error learning, and the mental task used, was the game of "Noughts and Crosses", sometimes known as "tic-tac-toe", this game represent a sequential decision process. Michie argued that a computer program could be enabled to improve its performance through its own accumulating experience [1].

In 1968 the matchbox machine was known as "Boxes" [2], first this project was undertaken as a "fun project", but there were more serious intentions to demonstrate the principle that states: "it may be easier to learn to play many easy games than a difficult one" [1]. Consequently, it may be advantageous to decompose a game into a number of mutually independent sub-games even when relevant information could be put out of reach in this process. The principle is related to the method of subgoals in problem-solving [3] but different in one

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 205-215

Received 03/03/07
Accepted 09/04/07
Final version 22/04/07

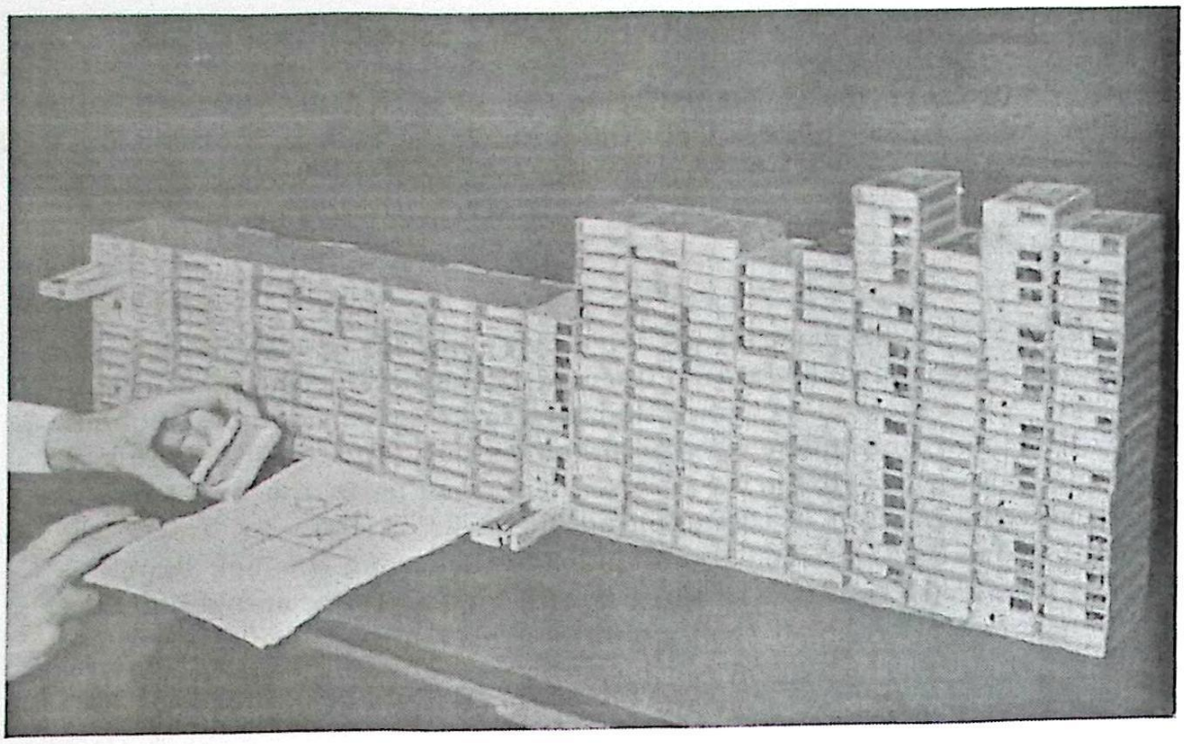


Fig. 1. The original matchbox machine known as "MENACE".

fundamental aspect: subgoals are linked in series, while sub-games are played in parallel.

The Boxes algorithm also was used in adaptive control problem in particularly the double pendulum [2]. In the adaptive control situation, where the states variables are real numbers, the large game is infinitely large, and then the sacrifice of information entailed in boxes approach is correspondingly extreme.

In this article, we used the one of the main ideas of the Boxes algorithm such as; learn to play many easy problems than a difficult one. This means that would be easier to learn the behavior of small parts of the time series than learn the whole behavior of the time series. Each small part is contained in the so-called "boxes" and they provide information about the amplitude behavior. In this work, we use this idea in order to track the dynamics of some time series with different behaviors.

The outline of this article is the following: firstly in section 2, we define the concept we used for time series and the approach of the general ideas of Boxes algorithm for time series tracking, in section 3, we explain the logic used and the implemented methodology; in section 4 we explain the obtained results and finally in section 5 some conclusions and future work are given.

2 Implementing the Boxes Algorithm to Tracking Time Series

We used the main idea from the Boxes algorithm explained before of divide the problem into small parts. Instead of matchboxes we used intervals which contain information about the amplitudes of the time series in order to track it. We are interested in the study of the time series with new techniques to learn its dynamics.

A time series is a sequence of data points, measured typically at successive times. The time series have information about the independent variables of a system which determines its dynamics. In other words, a time series is a sequence of values over the time of a system $x(t)$ which registers a sequence of experimental values [4][6][5]

$$x(t_1), x(t_2), x(t_3), ..., x(t_n) \tag{1}$$

for some interval $t = n$ with $t_0 < t_1 < ... < t_n$ [4], see Fig. 2 to observe an example of a time series.

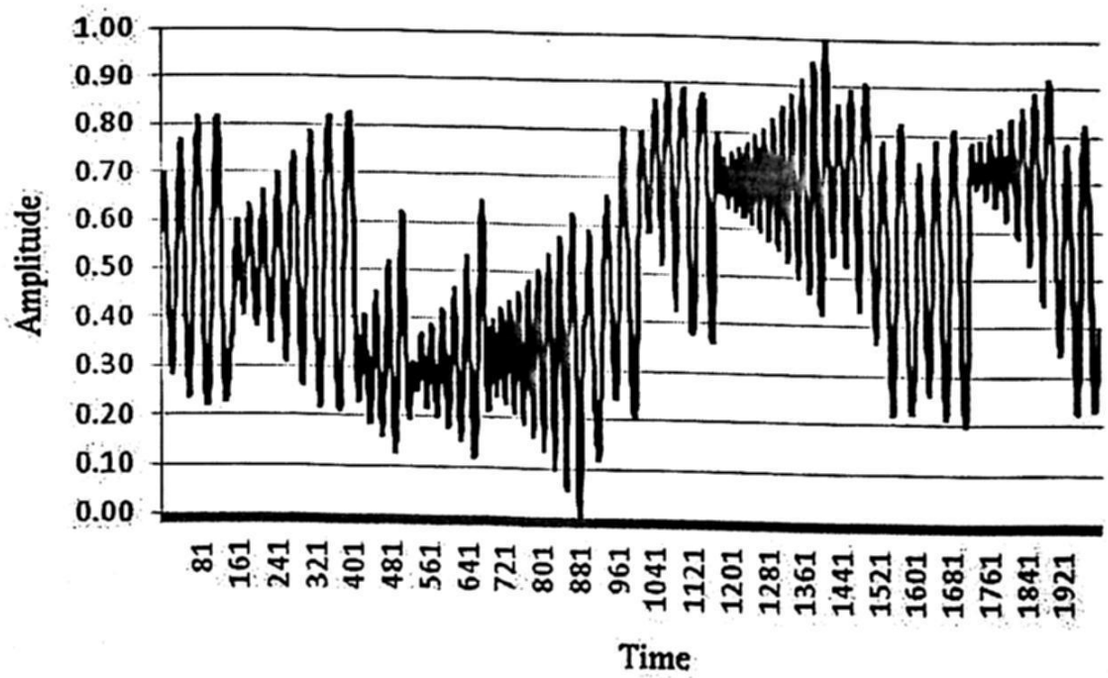


Fig. 2. Example of Lovaina time series.

In this work, we demonstrate that it is easier to learn the behavior of small amplitude intervals of the time series than trying to learn the behavior of the whole time series. That is because, the small intervals provided information from some specific amplitude of the whole time series and that allows a better learning of the behavior of it. We use this idea to generate a time series that tracks the behavior of the original one and that is what we call time series tracking, i.e., we

designed an algorithm that imitate the dynamics of a desired time series. The difference between our approach and the divide and conquer algorithm is that the last one works by recursively breaking down a problem into two or more sub-problems of the same (or related) type, until these become simple enough to be solved directly. The solutions to the sub-problems are then combined to give a solution to the original problem [8].

During the implementation of this approach we use a pseudo-random number generator therefore, this is a randomized algorithm or probabilistic algorithm. This algoritm employs a degree of randomness as part of its logic. In common practice, this means that the machine implementing the algorithm has access to a pseudo-random number generator. The algorithm typically uses the random bits as an auxiliary input to guide its behavior, in the hope of achieving good performance in the "average case". Formally, the algorithm's performance will be a random variable determined by the random bits, with (hopefully) good expected value; this expected value is called the expected runtime. The "worst case"is typically so unlikely to occur that it can be ignored [7].

3 On the Approach for Tracking a Time Series

As we mention before the idea behind this approach is related with a principle of subgoals in order to solve a complex problem by dividing the main problem into many subproblems that are sequentially linked but at the same time these subproblems are solved in parallel. Therefore, in this article we are using that principle to analyze the amplitude from the time series dividing it into intervals as is shown in the Fig. 3.

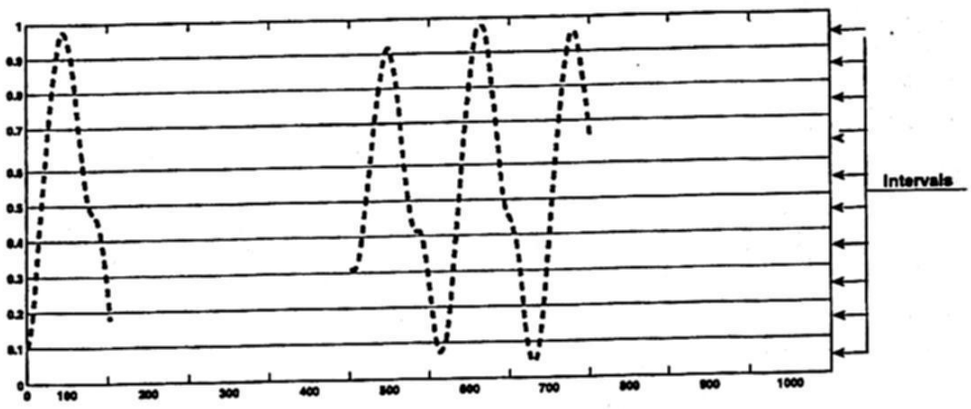


Fig. 3. Example of how we obtain new information from the time series. Additionally we can observe the division of the time series into ten intervals with the same width

The methodology used in this work follows the next steps:

- Normalize the original time series into the interval $[0, 1]$.

- Divide the normalized time series in eight intervals.
- Generate from a random variable the time series that tracks the dynamics of the original one.
- Compute the tracking error between the original time series vs. the computed.

The above steps can be exemplified in the next Figure 4.

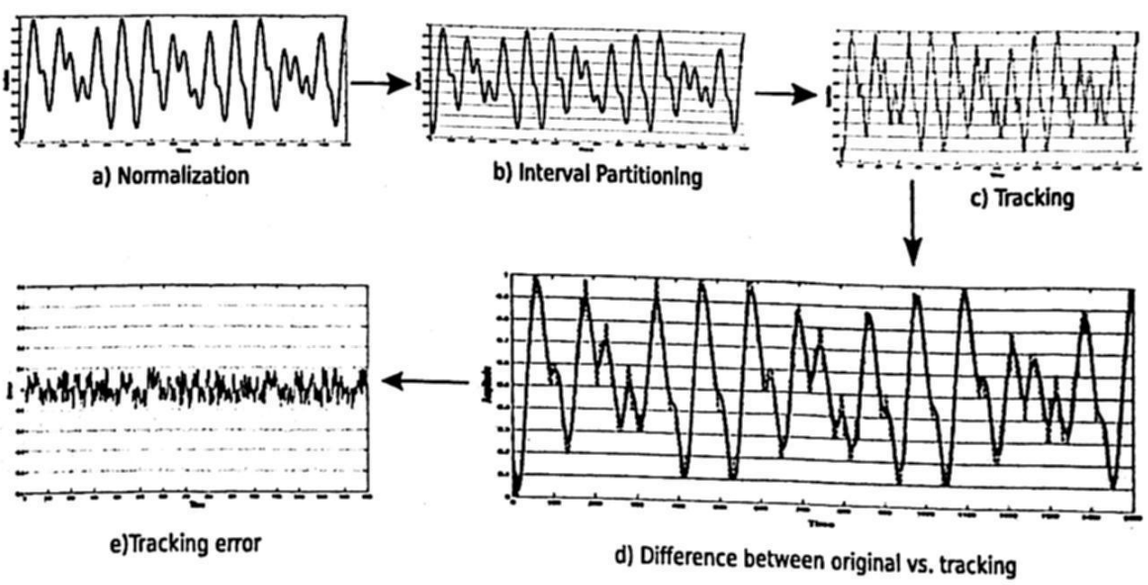


Fig. 4. Example of the metodology used for tracking time series. a) Normalization of the time series, b)Division of the original time series, c) Computation of the tracking time series, d) Difference between the both time series, e) Compute of the tracking error.

3.1 Normalization and Division

Broadly, normalization (also spelled normalisation) is any process that makes something more normal, which typically means conforming to some regularity or rule, or returning from some state of abnormality. In this approach normalization is very important due to the fact that we want to compare two or more time series from differents behaviors to prove the performance of this technique.

The following step is the division, this means the decision of how many intervals we are going to consider to divide all the time series that will analyze. From the mathematical definition an *interval* $[x]$ is a conected subset of $\mathbb{R}$. Even when the interval is not closed, we shall keep to the notation $[x]$. The *lower bound* $lb([x])$ of an interval $[x]$, also denoted by $\underline{x}$ is defined as

$$\underline{x} = lb([x]) \triangleq \sup\{a \in \mathbb{R} \cup \{-\infty, \infty\} \,|\forall x \in [x], a \leq x\} \tag{2}$$

Its *upper bound* $ub([x])$, also denoted by $\overline{x}$, is defined as

$$x = ub([x]) \triangleq \inf \{b \in \mathbb{R} \cup \{-\infty, \infty\} \,|\forall x \in [x], x \leq b\} \tag{3}$$

Thus, $\underline{x}$ is the largest number on the left of $[x]$ and $\overline{x}$ is the smallest number on its right. The *width* of any non empty interval $[x]$ is

$$w([x]) \triangleq \overline{x} - \underline{x} \tag{4}$$

In order to tests our approach we choose arbitrary the following eight intervals:

$$[-0.001, 0.125], [0.125, 0.25], [0.25, 0.37], [0.37, 0.5]$$

$$[0.5, 0.625], [0.625, 0.75], [0.75, 0.875], [0.875, 1] \tag{5}$$

Where the $lb([x])$ are open intervals and the $ub([x])$ are closed intervals.

3.2 Generation of the Time Series Tracking and its Error

The goal of this subsection is to find and extract information from the amplitude of the time series. In order to do that, we generate a new time series that tracks the behaviour of the original time series. The procedure we implemented is briefly shown in Fig. 5.

```
1 for i=1 to lastPointTimeSeries -1
2    x(i)=actualPoint
3     xnext(i+1)=nextpoint
4     if x(i)>lb([x]) and x(i)<ub([x])
5         if this is the first time this interval is visited then
6                 compute randPoint=rand([lb([x])], [ub([x])]
7                 compute the error= x(i)-randPoint
8         else means that this interval was visited previously then
9                  previosError=error(i-1)
10                 if previosError>0.0009 then
11                      newlb([x])=randPoint
12                      newub([x])=ub([x])
13                      compute randPoint=rand([newlb([x])], [newub([x])])
14                      compute the error=x(i)-randPoint
15                 end if
16        end if
17    end if
18 end for
```

Fig. 5. Main procedure to extract information from the original time series.

Firstly, we will start to moving a pointer from the first point of the original time serie to the last point (lines 1-3). Then we do the validation of the interval

and only the points from the time series that belongs to an specific intervalo ($[x]$) were considered in its analysis (lines 4-17). Thereafter the lines 5 to 8 only are visited when the first points belonging to certain interval enter for first time, because the lines 8 to 16 represent the training to get optimal rand numbers to future points. In line 9 there is an interesting validation, we notice that if we define a minimal error as in this case 0.0009 we obtaint better results than just keep it trainig without this validation and we define this error because it give us best results. If some points revisits the same interval and enter to line 11 means that we need to redefine the *lower bound* $lb([x])$ of the interval ($[x]$) in order to enclose the new interval $([newlb([x])], [newub([x])])$ and obtain a better result than the last computed.

The measure that we implemented in order to verify the performance of this approach was the root mean square error (RMSE) which is a frequently-used measure of the difference between values predicted by a model or an estimator and the values actually observed from the thing being modeled or estimated. In this particular case we used to measure the tracking error between the original time series and the computed one. The RMSE is defined

$$\sqrt{\frac{\sum_{i=1}^{n}(x_o - x_t)^2}{n}} \tag{6}$$

where x_o means the original points from the time series, x_t means the tracking series and n means the total number of the both time series [5].

4 Results

In order to verify the performance of this algorithm, we probed it with different time series such as: periodic, quasiperiodic, chaotic, complex and stochastic systems [5]. The data base we used to the time series tracking was taken from the reference [6] and is briefly shown in the Fig.6.

The software used in our simulations was Matlab 7.0®and the specification of the hardware we used is a CPU Pentium 4 with 3.0 GHz and 512 MB of RAM.

Some of the numerous experiments we performed are showed in the Fig. 7 to 10, and we are going to briefly explain each of the figures. Firstly in fig. 7 a) we depicted the zoom of the last 200 points of both time series, in order to see the difference between the original time series and the tracking one and in b) we can observe the errors at any time betwen the original point and the computed one obtained by the proposed algorithm and also we obtain the tracking error using the measure RMSE defined in the equation 6 and the error obtain of the tracking the behaviour from the Sine time series is equal to 0.0007 meaning that the tracking time series is resemblance to the original one as we can observe in that figure.

As we can see from Fig. 6 the time series plotted in Fig. 8 has a complex behavior which means that displays variation without being random and as can

Time Series	Dynamical Behavior
Sine Vanderpool	Periodic
Qperiodic2	Quasiperiodic
Mackey-Glass (*) Logistic Lorenz Rossler Ikeda Henon Cantor Tent	Chaotic
Laser Down Jones Kobe HIV DNA El niño Lovaina(*)	Complex
Browmian Motion White Noise(*)	Stochastic

Fig. 6. Some of the time series used to prove our technique. The time series marked with (*) are the ones used to obtain the experimental results showed in this paper.

be seen in this figure the difference between the original time series and the tracking one is minimum and this observation could be quantify by calculating the RMSE wich is equal to 0.00093. It is important to mention that the plotted points in this figure are the 300 last points in order to see the error between both series.

In the Fig. 9 we observe the last 200 points of the Mackey-Glass time series vs. the tracking one. This time series has a chaotic behaviour which means that it describes the behavior of certain nonlinear dynamical systems that under certain conditions exhibit dynamics that are sensitive to initial conditions (popularly referred to as the butterfly effect). As a result of this sensitivity, the behavior of chaotic systems appears to be random, because of an exponential growth of errors in the initial conditions. Despite of its behavior the results indicate that it is possible to track its dynamics and the RMSE obtained is equal to 0.018.

Finally, in Fig. 10 we observ the results obtained from the White Noise time series wich has a stochastic behavior. A stochastic process is one whose behavior is non-deterministic in that a state does not fully determine its next state. Therefore, in this figure we observe that the tracking is very accuracy due its behavior with a RMSE equal to 0.0024.

All the result indicates that it is possible to tracking time series from different behaviors obtaining in all the cases a very low error between the original and the time series tracking i.e., the RMSE.

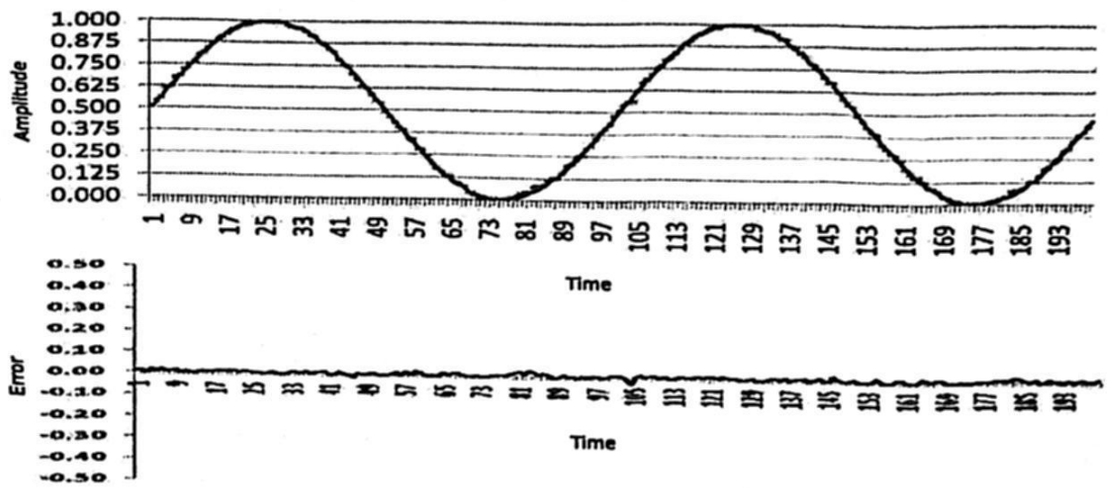


Fig. 7. A) Results of the Sine time series, which has a periodical behaviour. The continuos series (black) corresponde to the original points and the dotted series (gray) belongs to the tracking series. B)The Error tracking is showed

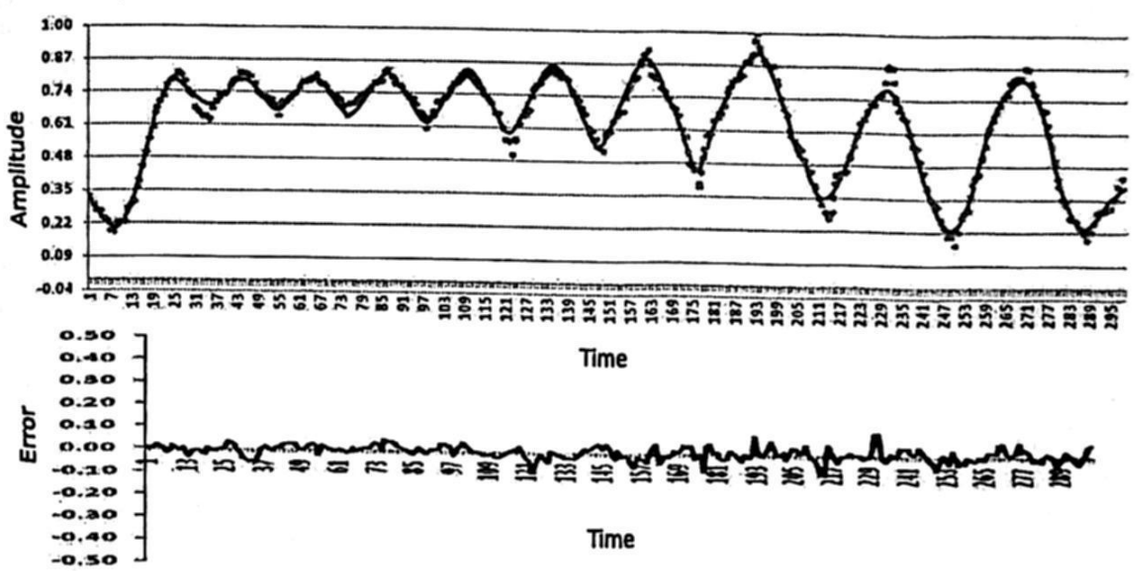


Fig. 8. A) Results of the Lovaina time series which is characteristic of complex behaviour. The original series es the continous one (black)and its tracking is the dotted (gray). B) The tracking error.

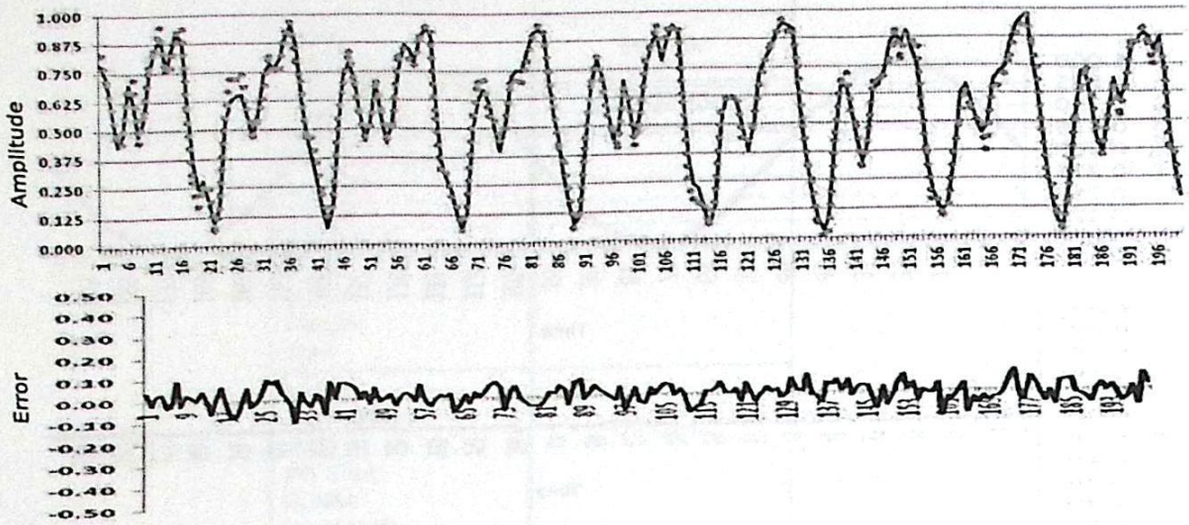


Fig. 9. A) Results of Mackey-Glass time seires, which has a chaotic behaviour. The continous series (black) correspond to the original points and dotted series (gray) belongs to the time series tracking. B) The tracking error is showed.

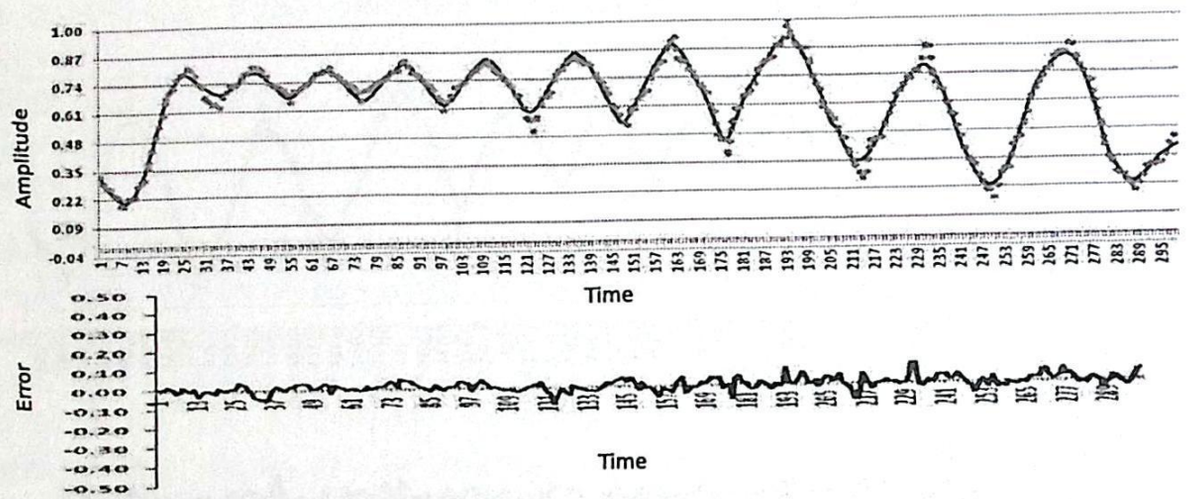


Fig. 10. A) Results of White Noise time series, which has a stochastic behaviour. The continous series (black) correspond to the original points and the dotted series (gray) belongs to the computed series. B) The tracking error is showed.

5 Conclusions

We have presented in this paper a new approach for time series tracking of different behaviors from an easiest time series (periodical) to a more complex behavior (chaotic). The tracking errors obtained with our technique demonstrated that the time series tracking has a high accuracy.

In the literature we found very few algorithms for time series tracking which makes a little difficult to compare our technique to others. Nevertheless, we consider this new approach a very accuracy one, because of the tracking errors that this technique obtains.

We think that some of the applications of the time series tracking are: tracking the trajectory of airplanes, reproduce music from the known music sheet among others.

At this moment we are working on the formalization of some aspects such as how many intervals is the best for any dynamic behavior of time series because in this work we used eight intervals determining it in an arbitrary way.

Using the information obtained with the time series tracking we can expand these results to a more difficult problem of time series namely, prediction. It is important to keep in mind that any natural phenomena can be able to be represented as a time series, and then it is possible to apply this approach in several time series tracking.

References

1. Michie, D.: Experiments on the mechanization of game-learning Part I. Characterization of the model and its parameters. Computer Journal **6** (1963) 232-236
2. Michie, D., Chambers, R. A.: Boxes: An Experiment in Adaptive Control. eds. E. Dale and D. Michie. Machine Intelligence 2. (1968) 125-133
3. Newell, A., Shaw, J.C., Simon, H.A.: A variety of intelligent learning. eds. C Marshall, Yovits and Scott Cameron. London: Pergamon : in a general problem solver in Self-organizing Systems. (1960). 153-189
4. Ramírez-Amaro, K., Figueroa-Nazuno, J.G.: Empleo de la Técnica Mapa Recurrente en el Análisis de los Índices de Teleconexión Atmosféricos. México, D.F. Centro de Investigación en Computación, I.P.N. (2006). Reporte Técnico. ISBN: 970-36-0330-0
5. Kantz, H., Schreiber, T.: Nonlinear Time Series Analysis. Cambridge : Cambridge University Press (1997)
6. Bautista-Thompson, E.F.: Medición de la Predictibilidad de las Series de Tiempo: un Estudio Experimental. México, D.F. Centro de Investigación en Computación. (2004). Ph. D. En Ciencias de la Computación.
7. Motwani, R., Raghavan P.: Randomized Algorithms. Cambridge University Press, New York (NY) (1995)
8. Frigo M., Leiserson C.E., Prokop H.: Cache-oblivious algorithm. Proc. 40th Symp. on the Foundations of Computer Science (1999).

A Tool for Long-memory Analysis in Computer Network Time Series

Julio C. Ramírez Pacheco[1], Deni Torres Román[2], and Joel Trejo Sanchez[1]

[1] Universidad del Caribe, SM 78, Manzana 1, lote 1, 77528, Cancún, QRoo, México
jramirez@ucaribe.edu.mx, jtrejo@ucaribe.edu.mx
[2] CINVESTAV Unidad Guadalajara, Av Científica 1145, 45010
Col. el Bajío, Zapopán, Jalisco, México
dtorres@gdl.cinvestav.mx

Abstract. The paper presents a novel software tool for *Hurst-index* estimation in *self-similar* and *long-range dependent* computer network time series. The tool, named *Variance Analyzer*, is based on the aggregated variance algorithm with tuned cut-offs. A comparison with *Selfis*, a similar tool for *long-memory*, using different dependence characteristics(*fGn* and *fDn*), shows that *Variance analyzer* presents better accuracy, time of convergence and faster estimations. Similar results are also obtained when using well-known real LAN traffic. The sources of inaccuracies in the algorithm are identified and the correct tuning is proposed.

1 Introduction

Computer network traffic's non-standard behavior is well studied and has been observed in several network configurations(LANs, WANs, VBR traffic, etc) [1] [2] [3] [4] [5]. *Extremes*, *heavy-tails*, *self-similarity* and *long-range dependence* are present in *delay*, *delay jitter*, file size, transmission times and aggregate traffic traces. The presence of these phenomena have a deleterious impact on computer networks' performance affecting the *quality of service* of applications [6] [9] [13]. Thus, an important problem in these traces is to correctly fit a model and then to efficiently quantify the degree of non-standard behavior(*tail-index* or *Hurst index* estimation) for the particular selected model. Once the model and the estimation is performed, the next step is to take actions in order to improve *quality of service* degree and the overall network performance. To accomplish the estimation, several algorithms have been proposed, each algorithm presents varying degrees of accuracy and time-domain or frecuency-domain properties [12] [7] [14] [13]. In this paper, a novel software tool for *self-similarity* and *long-range dependence* analysis in computer network time series is presented. The C++ based tool, named *Variance Analyzer*, is based on the time-domain aggregated variance algorithm. A comparison procedure against *Selfis*, a similar tool for *long-memory*, is accomplished. The comparison procedure is performed using several synthetic *fractional Gaussian noise* and *fractional differencing noise* time series with known *Hurst* exponent. Finally, the use of well-known real LAN traffic is

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 217-227

Received 23/02/07
Accepted 08/04/07
Final version 20/04/07

used. The organization of the paper is as follows. Section 2 discusses the measurement methodology in computer network traffic studies and the mathematical theory behind *self-similar* and *long-memory* processes. Section 3. provides description of the aggregated variance method for *Hurst index* estimation and comments on the sources of inaccuracies. Section 4. presents *Variance analyzer* characteristics and functionality. The comparison procedure and results is presented in section 5. Finally, section 6 concludes the paper.

2 Measurement Methodology and Mathematical Theory

In order to study computer network traffic's characteristics and its effect on network performance, some measurement of real traffic must be done. The result of the measurement is a trace from which a mathematical model is fitted. Usually, there is a relationship between network performance and some parameter of the selected mathematical model, therefore, accurate parameter estimation is important. The paper assumes that the trace fitted model is either *self-similarity* or *long-memory*. Next, we describe the measurement procedure in network performance studies and the mathematical models used in the paper. In the next section we cover parameter estimation of the assumed models.

2.1 Measurement Methodology

The first step in a network performance study is to obtain a trace from a measurement point. The measurement point could be a point in a LAN, WAN, link and path. A well known trace could also be generated and its behavior in a link or path could be an indicator of computer network performance. Note that in general, there are several ways to obtain a representative trace that can be used for network performance studies. The paper concentrates on the study of a trace representing the number of bytes/packets/bits per time unit on a measurement point. The trace, usually contains several types of traffic coming from different sources and with different *quality of service* characteristics. This trace is sometimes called the aggregate traffic trace or the traffic rate process and can be obtained from any computer network. Formally, let X_i represent the number of bytes/packets/bits for time period $(\tau_i - \tau_{i+1})$, then the trace $X = (X_i, i \in \mathbb{Z}+)$ contains the number of bytes/packes/bits for time periods $\{(\tau_i - \tau_{i+1})\}_{i \in \mathbb{Z}+}$. Note that the trace X represents a discrete-time stochastic time series that can be analyzed by probabilistic means. It has been shown that *self-similar* and *long-memory* processes model well the behaviour of the traffic rate process [1] [2] [4]. Description of *self-similar* and *long-memory* processes including its relationship is described next.

2.2 Self-similarity

Intuitively, *self-similarity* means that the properties(e.g. correlation structure, density) of an object(e.g. a time series) are mantained independently of scaling in

time and/or space. For traffic modelling purposes, the interest is in discrete-time statistical second-order *self-similar* processes with some form of stationarity. A discrete-time stationary stochastic process, $X = (X_t, t \in \mathbb{Z}+)$, is said to be second-order *self-similar*, with *self-similarity* parameter $H = 1 - \beta/2$, called the *Hurst* parameter, if its autocorrelation function $\rho(k), k \geq 1$ follows

$$\rho(k) = \frac{1}{2}((k+1)^{2-\beta} - 2k^{2-\beta} + (k-1)^{2-\beta}) = g(k), \tag{1}$$

where $\beta \in (0,1)$ and $H \in (1/2, 1)$. Equation (1), implying same correlation structure in all time scales, is too strict to model network traffic, therefore, an asymptotic behaviour giving rise to equation (1) is mostly used. Let $X_a = (X^{(m)}(k), k \geq 1)$ be the aggregated process of level m, obtained by appliying $X^{(m)}(k) = m^{-1} \sum_{t=(k-1)m+1}^{km} X(t)$ to the original time series $X = (X_t, t \in \mathbb{Z}+)$, then a discrete-time process is said to be asymptotic second-order *self-similar* if the aggregated process' autocorrelation function behaves asymptotically as

$$\rho^{(m)}(k) \sim \frac{1}{2}((k+1)^{2-\beta} - 2k^{2-\beta} + (k-1)^{2-\beta}), \tag{2}$$

as $m \to \infty$, $\beta \in (0,1)$ and $H \in (1/2, 1)$. Equation (1) implies that $\rho^m(k) = g(k), \forall m \geq 1$ and equation (2) implies that $\rho^m(k) = g(k)$ only aymptotically.

2.3 Long-memory

Intuitively long-memory or long-range dependence means that correlations between distant points in time of a series X_t are non-zero. This point relationship can occur only when the autocorrelation function behaves hyperbolically as opossed to exponentially. Long-range dependence in a stationary stochastic process, $\{X_t\}_{t \in Z_+}$ occurs when the lag k autocorrelation function in X_t, $\rho(k) : k \geq 1$ satisfies the following asymptotic behaviour

$$\rho(k) \sim c_p k^{-\beta}, \tag{3}$$

where $c_p > 0$ is a constant and $0 < \beta < 1$. Equation (3) implies that the sum of the autocorrelation function of X_t is not bounded, i.e., $\sum_{k=0}^{\infty} \rho(k) = \infty$ and the spectral density having a pole at zero, i.e., $f(\lambda) \sim c_f \mid \lambda \mid^{\beta-1}$ as $\lambda \to 0$. Another interpretation is that the correlations of a LRD process decay slowly in time, thus, giving rise to non-summability of the correlations.

2.4 Self-similarity and Long-memory Relationship

Self-similarity and long-range dependence are closely related concepts. An asymptotic self-similar process is defined according to equation (2), now let $k \to \infty$, then

$$\rho^{(m)}(k) \sim H(2H-1)k^{-\beta}. \tag{4}$$

Equation (4) implies that in the limit(as $k \rightarrow \infty$), an asymptotic *self-similar* process is *long-range dependent*. Similarly a *long-range dependent* process X_t can be constructed by the increment process of a *self-similar* process, i.e., $X_t = (Y_t - Y_{t-1}, t = 1, 2, ..)$, e.g., *fractional Gaussian noise* is obtained from the increment of a *fractional Brownian motion process*. For more information on *self-similar* and *long-range dependent* processes refer to [16] [8] [9] [15].

3 Parameter Estimation using Aggregated Variance

Once the mathematical model has been fitted to the measured computer network trace, estimation of some parameters characterizing the model is accomplished. Parameter estimation is important due to the relationship between parameter value and computer network performance [6] [9]. This relationship can be used for estimating computer network performance given some parameter value and for control algorithms' design in computer network performance applications [6]. Parameter estimation for *self-similar* and *long-memory* process is reduced to *Hurst-index* estimation which caracterizes completely their behaviour. Parameter estimation for these models can be done either in time-domain or frecuency-domain [12] [13] [14]. The paper concentrates on the time-domain algorithm named aggregated variance which is described next.

3.1 Aggregated Variance Method

Consider the aggregated series $X_a = (X^{(m)}(k), k \geq 1)$, obtained by dividing the original length N series in blocks of size m and computing the sample mean to each block, we take the sample variance to this series and obtain

$$Var(X^{(m)}) = \frac{1}{N/m} \sum_{k=1}^{N/m} (X_k^{(m)} - \bar{X})^2, \tag{5}$$

where $\bar{X}$ represents the original series sample mean. Equation (5) represents aggregated process' variance, mostly referred to as the aggregated variance. The aggregated variance method is based on the asymptotic behavior of the sample mean's variance in a discrete-time *self-similar* process $X_t, t \in \mathbb{Z}+$. The sample mean can be seen as the aggregated process of a discrete time series X_t, i.e., $E\{X_t\} = X^{(m)}$. Note that the aggregated series corresponds to the measured traffic rate process. Sample mean's variance decay, i.e., $Var(E\{X_t\})$, in *self-similar* processes behaves asymptotically as

$$Var(X^{(m)}) = Var(E\{X_t\}) \sim m^{2H-2}, \tag{6}$$

where $X_k^{(m)}$ is the aggregated process and H the *Hurst-index*. Note from this result that plotting the variance of the aggregated process $X^{(m)}$ versus m in log-log axes, for varying aggregation levels m, should result in a straight line with slope $2H - 2$. A least squares fit to the points in the plot should give the slope. Once the slope is estimated, the *Hurst-index* can easily be obtained [13].

3.2 Sources of Inaccuracies

Note that equation (6) gives an asymptotic behavior of the sample mean's variance, then, certain inaccuracies may appear due to the time series selected length. The longer the series the better the estimate. Additionally, due to the least squares estimate of the slope, the selection of the low and high end values of m, named cut-offs, affect the *Hurst-index*'s accuracy. A correct selection of these parameters(length and cut-offs)in any algorithm is then neccesary.

3.3 Accurate Estimation of the *Hurst-index*

Accurate estimations, as mentioned above, are obtained by the correct selection of the cut-offs and time series' length. Cut-offs and length selection, named tuning in this paper, is accomplished by first selecting the correct cut-offs and finally, based on the correct cut-offs, obtain a representative time series length. Cut-off selection is accomplished in two steps. First step is to select the high-end cut-off and the second involves determining the low-end cut-off value. High-end cut-off selection for an $H - index$ time series is accomplished by first selecting a fixed point in the x-axis, x_i, which is near the low-end regression value, then varying the points near the high-end regression value, x_j, $x_j \in (x_1, x_2)$. The high-end regression value which approximates better to the H value is selected as the high-end cut-off. Low-end cut-off selection is accomplished by varying the low-end regression values while maintaining the selected high-end cut-off fixed, as before, the low-end regression value which approximates better to the H value is selected as the low-end cut-off. Time series length selection is obtained via a cumulative analysis on the series. Cumulative analysis on a length N time series, X_t, $t = \{1, 2, \ldots N\}$, is obtained by first dividing the original length N time series in blocks of size $K \leq 1024$ and then estimating the *Hurst-index* for the series $\{X_i\}_{i=1}^{jK}$), $j = 1, 2, \ldots N/K$, i.e., $\hat{H_{jK}} = \Gamma(\{X_i\}_{i=1}^{jK})$, $j = 1, 2, \ldots N/K$, where $\Gamma(.)$ represents a *Hurst-index* estimation method. A plot of the estimated *Hurst-index* values $\hat{H_{jK}}$ versus j shows the behavior of the estimated *Hurst-index*. Usually, a stability region in the plot, is an indicator of the time series required length. The cut-off and time series' length selection procedures are performed by using synthetic long-memory traces, i.e., traces with well known *Hurst-index* values.

4 Variance Analyzer: A Tool for Long-memory

This section presents *Variance Analyzer* main features and functionality. It also describes briefly *Selfis*, a similar tool for *long-memory* analysis.

4.1 Variance Analyzer

Variance Analyzer is a novel C++ based software tool which estimates the *Hurst-index* using the aggregated variance method. The m values in the aggregated variance method in *Variance Analyzer* vary according to 10^x, $x =$

$0.1, 0.2, \ldots, 0.1(log_{10}(N))$, where N is the time series length. The selection of these values provides *Variance Analyzer* with better resolution than existing tools. An advantage of this is better accuracy but less convergence to long time series. The low-end and high-end cut-offs values are set to $10^{0.2}$ and $10^{2.2}$. These values were obtained using *fractional Gaussian noise* and FARIMA$(0, d, 0)$ synthetic long-memory traces. The time series length in these traces is set to 65536 points. Time series' *Hurst-index* estimation in *Variance Analyzer* is performed in two steps. First step involves the selection and automatic plotting of the text file. A requirement in this step is that the file should be in one-column format without spaces and comments. The non-conformance to this requirement causes *Variance Analyzer* to produce an error message. Once the file is plotted, a selection of a new file for analysis is possible. The second step involves estimating the *Hurst-index* of the selected and plotted time series. In this step, a plot of the regression points in the aggregated variance method is provided. Once the estimation is performed, *Variance Analyzer* provides functionality to return to step 1, i.e., to the orginial time series plot. As in the first step, a new file for analysis could be open. *Variance Analyzer* functionality is shown in the Petri net model of Figure 1. State **p1** is the initial state where *Variance Analyzer* is opened. State **p2** is the file open state, **p3** is the file plotted state and **p4** is the regression and *Hurst-index* estimation state. **T2**, **T5** and **T8** represent a non-valid file event and **T1**, **T4**, and **T7** represent a valid file event. Event **T3** is the plotting file event, **T6** corresponds to the estimation and regression event and **T9** represents the return event, i.e., the return to the original time series plot.

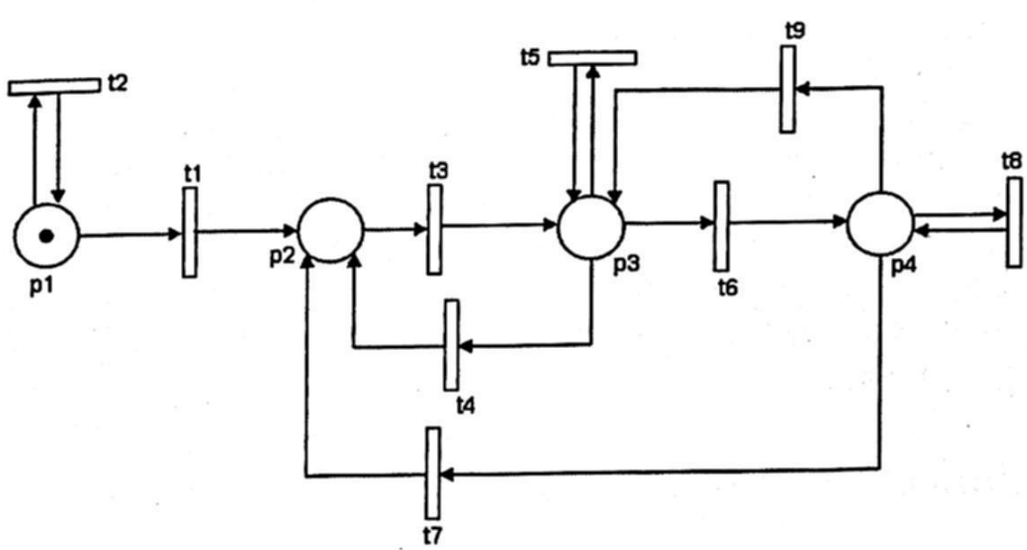


Fig. 1. *Variance Analyzer* Petri Net model

Figure 2 shows the user interface of *Variance Analyzer*. *Variance Analyzer* GUI consists of three main parts; the menu, the toolbar and the plotting area. The menu provides the user complete access to the functionality of *Variance*

Analyzer such as; file opening, program exit, *Hurst-index* estimation, etc. The toolbar provides the most often used functions such as file opening, *Hurst-index* estimation and the return button. The plotting area provides to the user the time series graphical representation and the regression points when applying the aggregated variance method.

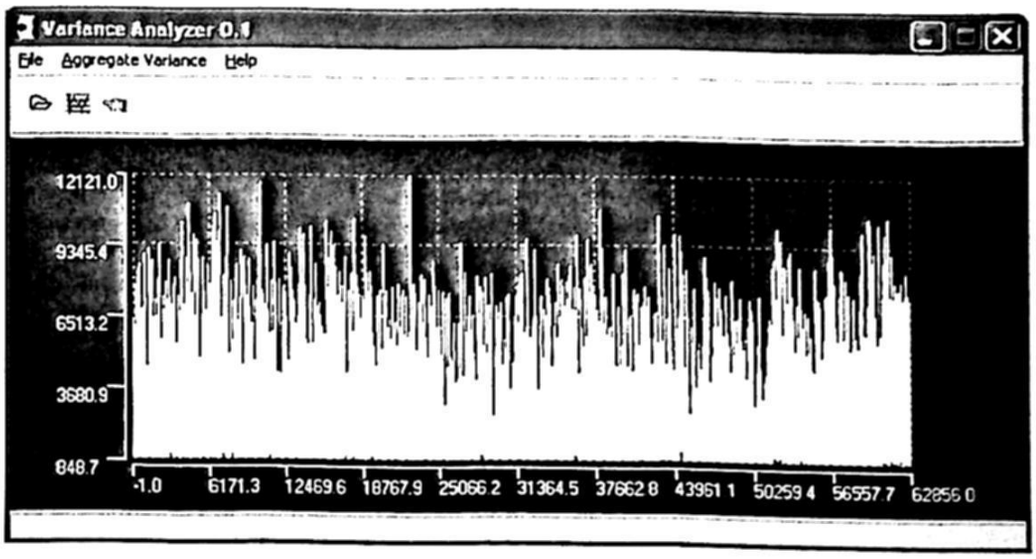


Fig. 2. *Variance Analyzer* User Interface

4.2 Selfis

Selfis is a well known software tool for *self-similarity* and *long-memory* analysis [10] [11]. *Selfis* is a Java-based software tool which estimates the *Hurst-index* using four time-domain methods and three frecuency-domain methods. It was noted that *Selfis* does not present the cut-offs used in the *Hurst-index* estimation. This paper is interested in the accuracy of *Selfis* aggregated variance method. *Selfis* aggregated variance implementation differs from that of *Variance Analyzer*, thus, our aim is to quantify the accuracy of each in estimating the *Hurst-index* for different synthetic and real *long-memory* traces. For more information on *Selfis* characteristics and functionality refer to [10] [11]

5 Comparison Procedure and Results

The comparison procedure of *Variance Analyzer* versus *Selfis* is presented in this section. Synthetic and real trace description is presented first, the comparison procedure is described next and finally, *Hurst-index* estimation using both tools is performed.

5.1 Long-memory Traces

In order to quantify the accuracy degree in *long-memory* analysis software tools, the use of traces with known *Hurst-index* are used. These traces are commonly named synthetic *long-memory* time series. The paper makes use of two synthetic *long-memory* time series types, namely *fractional Gaussian noise*(fGn) and FARIMA$(0, d, 0)$(*fractional differencing noise,* fDn) time series. fGn time series, are long-memory time series satisfying equation (1). fGn traces were created using modified Paxson's FFT algorithm with an asymptotic mean zero decay [17] [18]. Fractional differencing noise(FARIMA$(0, d, 0)$) time series, are series satisfying

$$X_i = \Delta^{-d}\epsilon_i,\ i \geq 1, \tag{7}$$

where ϵ_i are *iid* Gaussian random variables with zero-mean and Δ is the differencing operator satistying $\Delta = \epsilon_i - \epsilon_{i-1}$. The autocovariance function of this process satisfies

$$\rho(k) = C_e k^{2d-1},\ h \to \infty, \tag{8}$$

where $d \in (-1/2, 1/2)$ and $C_e = \pi^{-1}\sigma^2\Gamma(1-2d)\sin(\pi d)$. For large lags, the ACF for fGn and fDn has the same power decay, thus, $H = d + (1/2)$. fDn time series were created using $S+$ package. Finally, the application of well-known real LAN traffic traces are used. The traces are the well-studied an classical traces of [1] [2].

5.2 Comparison Procedure

The comparison process was performed by using the synthetic traces described above. A set of nine fGn traces with *Hurst-index* from 0.55 to 0.95 in increments of 0.05 were created. An identical set of traces for fDn were also created. The length of the series, both for fGn and fDn, was set to 65536 points. For the real trace case, the use of **AUG89.MB** and **AUG89.MP** LAN time series from Bellcore were studied. The length for these series is about 360000 points representing the number of bytes(**AUG89.MB**) and packets(**AUG89.MP**) per time unit in a LAN environment.

5.3 Results

Table 1 shows *Hurst-index* estimations for *Variance Analyzer* and *Selfis* using fGn synthetic traces. Note that *Selfis* presents high bias for traces with *Hurst-index* 0.55 and $0.70 - 0.95$. The bias($\epsilon = H_{theoretic} - \hat{H}_{estimated}$) in *Selfis* tool for these traces is $\epsilon \geq 0.035$. Unlike *Selfis*, *Variance Analyzer* presents minimum bias estimates for the *Hurst-index* in the interval $0.55 - 0.90$. Note that *Variance Analyzer* estimations are more accurate than *Selfis* for the fGn case. Table 2 shows the *Hurst-index* estimations for both tools when using FARIMA$(0, d, 0)$ long-memory time series. As can be noted from the table *Selfis* presents accurate estimates only for the fDn traces with *Hurst-index* 0.60 and 0.85. *Variance*

Table 1. *Hurst-index* estimations using fGn traces

	Selfis	Variance Analyzer
Hurst-index		
0.55	0.515	0.5526
0.60	0.586	0.5906
0.65	0.673	0.6507
0.70	0.594	0.7019
0.75	0.696	0.7391
0.80	0.720	0.7845
0.85	0.795	0.8431
0.90	0.804	0.8739
0.95	0.798	0.9088

Analyzer, unlike *Selfis*, presents accurate estimation in the $(0.55, 0.85)$ interval. From the results it is said that *Variance Analyzer* presents more accurate estimations of the *Hurst-index* than *Selfis* both for fGn and fDn *long-memory* synthetic traces. The attention is now turned to the analysis of real LAN computer network time series. The study of these traces is important for testing the capability of algorithms in a real environment. Figure 3 shows *Hurst-index* estimations for **AUG89.MB** and **AUG89.MP** LAN traces. As can be seen from the table, *Selfis*, presents problems for long time series and is unable to open these types of traces. **AUG89.MB** and **AUG89.MP** length is 360000 points. Unlike *Selfis*, *Variance Analyzer* is capable of opening this file and presents accurate estimations for these traces. From this study and the above, it is seen that *Variance Analyzer* presents better accuracy either for synthetic and real *long-memory* time series.

Table 2. *Hurst-index* estimations using fDn traces

	Selfis	Variance Analyzer
Hurst-index		
0.55	0.466	0.5499
0.60	0.584	0.5890
0.65	0.570	0.6341
0.70	0.665	0.6921
0.75	0.648	0.7339
0.80	0.765	0.7756
0.85	0.841	0.8438
0.90	0.730	0.8674
0.95	0.839	0.9050

Table 3. *Hurst-index* estimations using real LAN traces

	Selfis	Variance Analyzer
Trace(*Hurst-index*)		
AUG89.MB(~ 0.80)	*NotOpened*	0.8166
AUG89.MP(~ 0.90)	*NotOpened*	0.8662

6 Conclusions and Future Work

A tool for *long-memory* and *self-similarity* analysis for computer network time series was presented. The *Hurst-index* estimation tool, named *Variance Analyzer*, is based on the aggregated variance algorithm with tuned cut-offs. The sources of inaccuracies in the aggregated variance algorithm were identified and the correct selection of the low and high end cut-offs was proposed. A comparison procedure of *Variance Analyzer* versus *Selfis* showed that *Variance Analyzer* presents better accuracy and minimum-bias estimates of the *Hurst-index*. The comparison was performed by using known *Hurst-index* and real LAN time series. *Variance Analyzer* robustness to long time series was also accomplished. Based on this, *Variance Analyzer* should be the tool of choice when analyzing time series via the aggregated variance method for fGn and fDn-like time series. *Variance Analyzer* could also be employed for the analysis of other non-computer network time series, e.g., geological, hydrological, etc.

References

1. Leland, W. E., Taqqu, M.S., Willinger, W., Wilson, D. V.: On the self-similar nature of Ethernet traffic. Proc. ACM SIGCOMM '93, San Francisco CA. (1993) 183–193
2. Leland, W. E., Taqqu, M.S., Willinger, W., Wilson, D. V.: On the self-similar nature of Ethernet traffic(Extended version). IEEE/ACM Transactions on Networking **2** (1994) 1–15
3. Beran, J., Sherman, R., Taqqu, M. S., Willinger, W.: Long-range dependence in variable-bit-rate video traffic. IEEE Transactions on communications **43** (1995) 1566–1579
4. Crovella, M., Bestavros, A.: Self-similarity in Word-Wide-Web traffic: evidence and possible causes. IEEE/ACM Transactions on Networking. **5** (1997) 835–846
5. Paxson, V., Floyd, S.: Wide-area traffic: The failure of poisson modelling. IEEE/ACM Transactions on Networking. **3** (1995) 226–244
6. Park, K.: On the effect of traffic self-similarity on network performance. Proc. SPIE International conference on performance and control of network systems. (1997) 296—310
7. Taqqu, M. S., Teverovsky, V., Willinger, W.: Estimators for long-range dependence: An empirical study. Fractals. **3** (1995) 785–798
8. Beran, J.: Statistics for Long-memory Processes. New York, Chapman & Hall (1994)

9. Park, K., Willinger W.: Self-similar Network Traffic and Performance Evaluation. Wiley-Interscience (2000)
10. Karagiannis, T., Faloutsos, M., Molle, M.: A User-Friendly Self-similarity Analysis Tool. Special section on Tools and Technologies for Networking Research and Education, ACM SIGCOMM Computer Communication Review. (2003)
11. Karagiannis, T., Faloutsos, M.: SELFIS: A Tool for Self-similarity and Long-range Dependence Analysis. 1st Workshop on Fractals and Self-similarity in Data Mining: Issues and Approaches.(2002)
12. Taqqu, M. S., Teverovsky, V.: Semi-parametric graphical estimation techniques for long-memory data Athens Conference on Applied Probability and Time series analysis. **115** (1996) 420–432
13. Adler, R., Feldman, R., Taqqu, M. S.: A Practical guide To Heavy-Tails: Statistical Techniques and Applications. Boston, Birkhauser. (1998)
14. Kokoszka, P., Taqqu, M. S.: Parameter Estimation for infinite variance fractional ARIMA. Annals of Statistics. **24** (1996) 1880–1913
15. Lopez-Ardao, J., Lopez-Garcia, C., Suarez-Gonzalez, A., Fernandez-Veiga, M., Rodriguez-Rubio, R.: On the use of Self-similar Processes for Network Simulation. ACM Transactions on Modelling and Computer simulation. **10** (2000) 125–151
16. Tsybakov, B., Georganas, N.: Self-similar Processes in Communications Networks. IEEE Transactions on Infotmation Theory. **44** (1998) 1713–1725
17. Paxson, V.: Fast, Approximate Synthesis of Fractional Gaussian Noise For Generating Self-similar Network Traffic. Computer Communications Review. **27** (1997) 5–18
18. Rolls, D.: Improved Fast Approximate Synthesis of Fractional Gaussian Noise. Hawaii International Conference on Statistics and Related Fields. (2002)

Robotics and Control

Biologically-Plausible Reactive Behaviors for Robots and Virtual Characters

Abraham Sánchez L., Gerardo Garcia L. and René Zapata‡

Facultad de Ciencias de la Computación, BUAP
14 Sur esq. San Claudio, CP 72570
Puebla, Pue., México
‡ Robotics Department, University of Montpellier II
161 rue Ada, 34392 Montpellier Cedex 5 - France

Abstract. Reactive systems are a relatively recent development in robotics that has redirected artificial intelligence research. This new approach grew out of a dissatisfaction with existing methods for producing intelligent robotic response and a growing awareness of the importance of looking at biological systems as a basis for constructing intelligent behavior. This works addresses the modeling of the reactive behaviors for robots and virtual characters moving in unstructured and dynamic environments. This model is based on the definition of the interaction component (deformable virtual zone) of an internal state of the robot (or the character) and leads to avoidance-oriented control laws. Experimental results show the effectiveness of the proposed approach here.

1 Introduction

Collision avoidance has been an active research topic in Robotics. In general there have been two directions in this area: motion planning and reactive behavior. In motion planning, a collision-free path is planned an the robot is instructed to follow the path [7], [3]. In the later approach, collision avoidance is built as a reactive behavior of the robot. When the robot is close to an obstacle, it produces a repulsive force which will push the robot away and thus avoid collision [4].

Both approaches have advantages and disadvantages. The reactive behavior approach has a clear advantage in performance. In most cases it can be done in real time or near real time, even when the number of degrees of freedom (dofs) involved is large. Reactive methods can even avoid moving obstacles [10]. These methods suffer, however, from the local minima problem. While the motion planning approach does not suffer the local minima problem in general, it is usually too slow to be used in an interactive (changing) environment.

From path planning to trajectory control, the motion planning problem for robots has been thoroughly investigated in the case of structured environments. Moving among unknown or badly modeled environments, practically induces the necessity of taking unscheduled and dynamic events into account and reacting as the living beings would do. Therefore, reactive behaviors play a fundamental role when the robot has to move through unstructured and dynamic environments.

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 231-242

Received 09/02/07
Accepted 08/04/07
Final version 21/04/07

For the last fifteen years, the scientific community has been interested in the problem of reactive behaviors for collision avoidance in the domain of mobile robots [11]. This kind of algorithm is based on the definition of a protecting and deformable zone surrounding the robot (see next section).

In this trend, this work aims at providing a practical planner that considers reflex actions and PRM techniques to account for planning with changing obstacles. The paper is organized as follows. Section II gives an overview of the DVZ principle. Section III describes the reactive PRM approach for both holonomic and non-holonomic motion planning. A novel idea to solve the locomotion planning for virtual characters in dynamic environments is discussed in the section IV. Finally, the conclusions and future work are presented in section V.

2 Biologically-Plausible Reactive Behaviors

In many industrial, exploration or lab robotic tasks, a robot mainly has to perform some subtasks: (i) collision avoidance, (ii) target pursuit, (iii) localization, and (iv) path planning. The first three tasks obviously need perception capabilities, while the fourth one also does when the environment is subject to change (re-planning when a failure occurs).

The reactive control algorithm we use is based on the definition of a protecting and deformable zone surrounding the robot. The DVZ (Deformable Virtual Zone) is parameterized by the motion variables of the moving robot and can deform in the presence of distance information in the robot workspace. When an obstacle enters the sensor space, it induces a deformation of the DVZ that will be compensated by the robot motion controller. Therefore, the algorithm is a kind of 2-player game: the first one, i.e. the environment, induces undesired deformations; the second one, i.e. the robot controller, tries to rebuild the DVZ. This algorithm was first described in [11] for many applications and tested for robots moving in 2 dimensions [10], flying robots or autonomous submarines and also mobile manipulators [2]. This algorithm which was initially designed for obstacle avoidance, has two main advantages. First, the environment does not need to be a priori known, and second, the controller can take into account other constraints such as target pursuit, altitude maintenance, course control and so on. Figure 1 illustrates this general principle that will be described in the next paragraphs.

2.1 Mathematical Basis

The general framework for formalizing this principle is the category $\mathcal{D}$ of topologically equivalent sets of the $(n-1)$-dimensional unitary sphere $S^{n-1}(0,1)$ in $\mathbb{R}^n$. An object A of this category is related to the unitary sphere through an homeomorphism (imbedding of the sphere):

$$\delta_A : S^{n-1}(0,1) \rightarrow A \subset \mathbb{R}^n \tag{1}$$

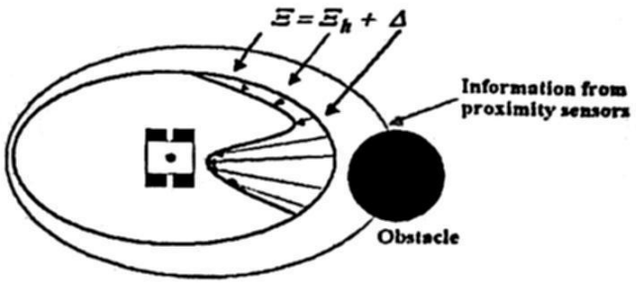


Fig. 1. DVZ principle in 2D.

The transformations between two objects A and B of $\mathcal{D}$ are the deformations obtained by the combination of the two defining homeomorphisms:

$$\delta_B \circ \delta_A^{-1} : A \rightarrow B \tag{2}$$

Let $R \subset \mathbb{R}^n$ be a convex rigid body, subset of the nD-space $\mathbb{R}^n$. The boundary ∂R of R can also be considered as the result of an imbedding of $S^{n-1}(0,1)$ in $\mathbb{R}^n$. We have $\partial R \in \mathcal{D}$ and:

$$\delta_R : S^{n-1}(0,1) \rightarrow \partial R \subset \mathbb{R}^n \tag{3}$$

Reciprocally, if f is an imbedding of ∂R in $\mathbb{R}^n$ with $E = f(\partial R)$, we can say that

$$f : \partial R \rightarrow E \subset \mathbb{R}^n \tag{4}$$

Any object $A \in \mathcal{D}$ separates $\mathbb{R}^n$ in two connected components, the interior $Int(A)$ of A and the exterior $Ext(A)$ of A. Therefore, we have $\mathbb{R}^n = Int(A) \oplus Ext(A) \oplus A$. A partial order is induced on $\mathcal{D}$ by the relation:

$$A \prec B \Leftrightarrow Int(A) \subset Int(B) \tag{5}$$

The rigid body R will represent a controlled robot moving among obstacles in $\mathbb{R}^n$. Any n-dimensional state vector characterizing the motion of R is denoted as a vector

$$\pi = [p_1\ p_2\ \ldots p_n]^T \tag{6}$$

Axiom 1 *We assume that the robot R can be controlled by the derivative of this state vector. We note*

$$\phi = \dot{\pi} \tag{7}$$

Definition 1 *We define a DVZ of R as any imbedding Ξ of ∂R in $\mathbb{R}^n$ such that the relation $\partial R \prec \Xi(\partial R)$ holds. We have $\Xi(\partial R) \in \mathcal{D}$.*

Definition 2 *We define a controlled DVZ, Ξ_h, as a DV which depends on the state vector characterizing the motion of R:*

$$\Xi_h = \rho(\pi) \tag{8}$$

Definition 3 *Let $\mathcal{P} = (\Xi_h, \Xi)$ be a pair of two DVZ of R (the first one being a controlled DVZ) and such that $\Xi(\partial R) \prec \Xi_h(\partial R)$. We define the deformation Δ of the DVZ Ξ_h with respect to Ξ as the functional difference of Ξ and Ξ_h:*

$$\Delta = \Xi - \Xi_h \tag{9}$$

According to this definition, the deformation Δ is a one-one map that associates the vector $P - P_h$ to the point $M \in \partial R$, where $P = \Xi(M)$ and $P_h = \Xi_h(M)$. It can therefore be considered as a vector field defined on ∂R.

Axiom 2 *We assume that the robot can perceive distances in all directions of space. We also assume that the set of maximum distances that can be perceived by R and the set of actually perceived distance are two objects of the category $\mathcal{D}$, respectively named sensor boundary and information boundary (respectively denoted by Θ and Ψ), such that $\Psi \prec \Theta$. The deformation I of Θ with respect to Ψ is given by:*

$$I = \Psi - \Theta \tag{10}$$

The deformation I can also be considered as a vector field on ∂R.

Definition 4 *We define an uncontrolled DVZ, Ξ, as a DVZ which depends onf the sensor boundary deformation I:*

$$\Xi = \beta(I) \tag{11}$$

Let $\mathcal{P} = (\Xi_h, \Xi)$ be a pair composed of a controlled DVZ and an uncontrolled DVZ, the deformation Δ of the DVZ Ξ_h with respect to Ξ can be written:

$$\Delta = \Xi - \Xi_h = \beta(I) - \rho(\pi) \tag{12}$$

For a given point $M \in \partial R$, the deformation vector $\Delta(M)$ depends on the intrusion of proximity information $I(M)$, in the rigid body workspace and on the controlled DVZ Ξ_h.

By differentiating equation (12) with respect to time, we get:

$$\dot{\Delta} = -\nabla_\pi[\rho]\phi + \nabla_I[\beta]\psi \tag{13}$$

where ∇_ξ is the derivation operator with respect to the vectorial variable ξ and $\psi = \dot{I}$.

This equation can be rewritten as:

$$\dot{\Delta} = A\phi + B\psi \tag{14}$$

Variations in Δ are controlled by a 2-fold input vector $u = [\phi\ \psi]^T$. The first control vector ϕ, due to the robot controller tends to minimize deformation of the DVZ. The second one, ψ is unknown and induced by the environment itself (and could, at most, try to maximize these deformations). This equation will be referred as the main equation of the problem.

Once the main equation obtained, its integration (i.e., the obtention of a "good" control vector ϕ) can be computed in 2 steps:

1. Choosing the desired variation of this deformation as a function of the real deformation and its derivative:

$$\dot{\Delta}_{des} = -K_{prop}\Delta - K_{der}\dot{\Delta} \tag{15}$$

   where K_{prop} and K_{des} are heuristically chosen.
2. Computing the best control vector $\check{\phi}$ at time t obtained by inverting equation (14) after replacing the deformation derivative by its desired value $\dot{\Delta}_{des}$:

$$\check{\phi} = A^{\dagger}(\dot{\Delta}_{des} - B\hat{\psi}) \tag{16}$$

   where $A^{\dagger}$ is the inverse function (pseudo-inverse) of the linear function A and $B\hat{\psi}$ is an estimation of the second control vector ψ at time t obtained at time $t-1$:

$$B\hat{\psi}(t) = \dot{\Delta}_{measured}(t-1) - A\phi(t-1) \tag{17}$$

The control law (equation 16) tends to minimize the function $\|\dot{\Delta}_{des} - \dot{\Delta}\|$ in the least-squares sense. The ∞-dimensional functional equation 14 cannot, of course, be used directly. It is necessary to sample the sensor space in order to obtain an n-dimensional definition of the DVZ. This can be done by considering that the information vector has n dimensions (as many as the number of distance sensors). Equation 14 keeps its general form but all its entries are now matrices or vectors. Figure 2 shows some DVZ examples in 3D.

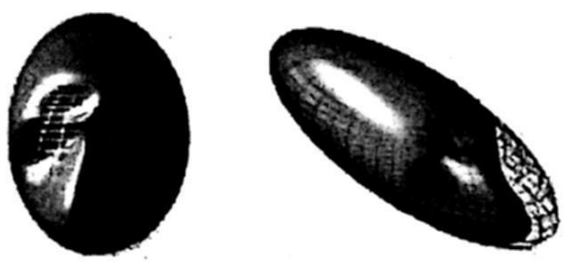


Fig. 2. Some DVZ examples in 3D.

In summary, collision avoidance is a 4-step process:

1. Measurement of the intrusion of information I as an n-dimensional vector (as many dimensions as the number of sensors).

2. Derivation of the deformation Δ and of its derivative $\dot{\Delta}$.
3. Estimation of the uncontrolled control vector $\hat{\psi}$.
4. Computation of the best control vector $\check{\phi}$.

3 Reactive Behaviors in Motion Planning

Robot motion planning has led to active research over the two last decades. In particular, probabilistic techniques have received a lot of attention in recent years [3]. They have proven to be effective methods that can be applied to different problems arising in fields as diverse as robotics, graphics animation, computational biology.

Probabilistic roadmap method (PRM) is a general planning scheme building probabilistic roadmaps by randomly selecting configurations from the free configuration space and interconnecting certain pairs by simple feasible paths. The method has been applied to a wide variety of robot motion planning problems with remarkable success. PRM planners have been originally designed for solving multiple-query or single-query problems.

Dynamic changes in the environment are very common in many motion planning applications such as planning for evolving industrial environments, navigation in real or in virtual worlds.

The adaptation of PRM planners to environments with both static and moving obstacles has been limited so far. This is mainly because the cost of reflecting dynamic changes into the roadmap during the queries is very high. On the other hand, single-query variants, which compute a new data structure for each query, deal more efficiently with highly changing environments. They however do not keep the information reflecting the constraints imposed by the static part of the environment useful to speed up subsequent queries.

The proposed approach integrates the lazy PRM planning method [1], [6] and the reactive control by DVZ [11] in the following way: a collision-free feasible path for the robot is calculated by the lazy PRM method, the robot starts moving (under the permanent protection of its DVZ), in the absence of dynamic obstacles, the control is performed by the lazy PRM method and does not require reflex commands. If there are dynamic obstacles in its path, the reactive method takes the control and generates commands to force the robot to move away from the intruder obstacles and gives back its DVZ to the original state.

In this point, the robot has lost its original path, and it is necessary to search for a reconnection path to reach its goal. The new path found is a single collision-free curve of Reeds & Shepp for non-holonomic robots or a straight-line path for holonomic ones. If the attempt of reconnection is successful, the robot executes its new path towards the goal. The new alternative path was obtained with the lazy PRM method by using the information stored in the current robot's configuration, but if a deformation appears, the processes are interrupted by reflex actions that forces the planner to go back to the previous state.

After a successful reflex action, the mobile robot recovers the intact state of its DVZ, but its initial planned path will be lost. The lazy PRM method

needs to have a path to push the robot to the goal and it will be necessary to provide a path for such aim. Due to the high computational cost of a complete replanning, the method will avoid it by executing a process that uses a single path to reconnect with the planned path.

If the reconnection attempts fails, it may happen that paths are blocked by many dynamic objects, or a moving object is parked obstructing the planned path. In this case, the planner executes the lazy PRM method (the initial configuration is the current configuration in the robot). The lazy PRM will be called several times until it returns a collision-free path. If after some attempts a collision-free path can not be found, the planner reports failure.

The algorithm can finish in three forms: i) the robot executes its path successfully, ii) the reflex action is not sufficient and a collision occurs, or iii) the robot does not find an alternative path to conclude its task.

In order to evaluate the performance of the proposed approach, we present some experimental results for car-like robots. The moving obstacles have a square form and move at constant velocity in straight line. Whenever they collide with another object they assume a new random direction in their movement.

Figure 3 shows an environment composed of four static obstacles and several dynamic obstacles moving randomly at the same velocity than the mobile robot. Figure 4 shows an environment with narrow passages.

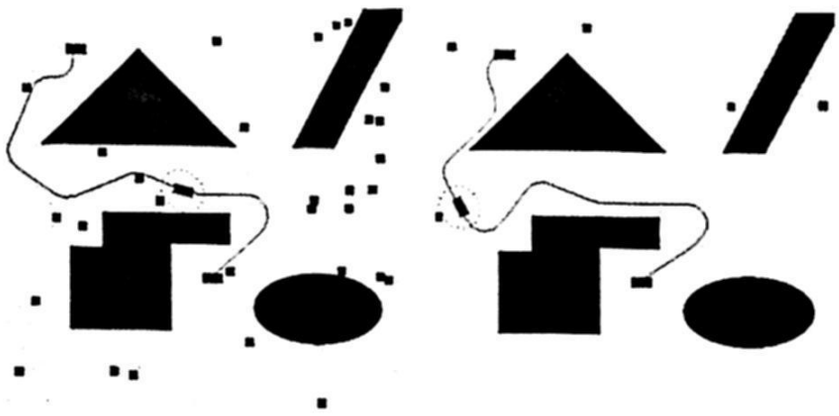


Fig. 3. An example of a query and its solution path (left) in an environment with 30 moving obstacles. The robot starts moving under the permanent protection of its DVZ (right), the scene contains 5 moving obstacles.

In fact, the method's performance can be considered satisfactory if it presents a fast planning phase, reflex actions based on sensors that do not require expensive algorithms, an effective process of reconnection performed in milliseconds, and a process of replanning that is executed if the Lazy PRM and DVZ' s parameters are appropriate. As mentioned in earlier sections, it can be considered that the methodology proposed here, includes these characteristics. The planning time is reduced due to the incomplete collision detector whose work is complemented with the robot's sensors during the path execution. On the other hand, the assignation of direction angles to the nodes that conform the shortest

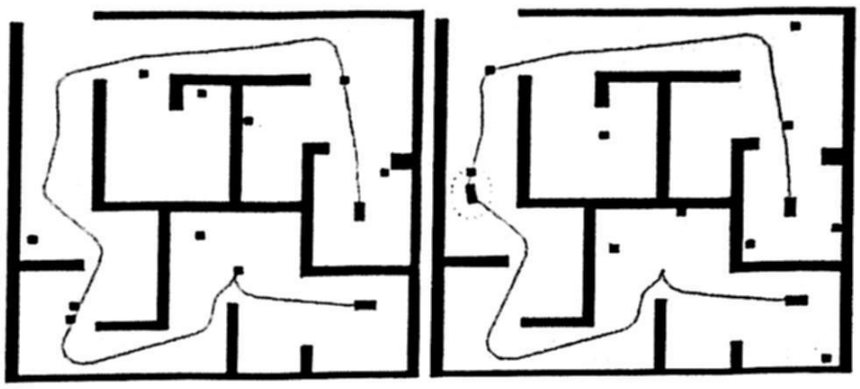


Fig. 4. An environment composed of some narrow passages. The scene contains 10 moving obstacles.

paths obtained by the algorithm A^*, produces curves that allow the algorithm to omit the optimization process (i.e., the smoothing process). With respect to the reconnection process, the paths obtained with the planner are conformed by a single Reeds & Shepp curve and based on the incomplete collision detector, making short the time and close to optimal the curves obtained with the algorithm. Since the reflex actions are provided by the DVZ method, it is possible to interrupt the reconnection and replanning processes if necessary, without incurring in bigger problems.

The proposed method in this section is general, even though we presented the case of motion planning for car-like robots. The DVZ's form is different, for manipulator robots we can use a cylinder.

4 Animating Reactive Motions

The synthesis of realistic human motion is a challenging research problem with broad applications in movies, cartoons, virtual environments, and games.

Due to the quality and realism of the result, the use of motion captured data has become a popular and an effective means of animating human figures [8]. However, since it is an inherently off-line process, there has been great interest in developing algorithms that are suitable for interactive applications. Designing appropriate control schemes can be difficult and only a limited number of methods consider reactive motions due to the presence of applied external forces [12], [5].

We propose to solve the locomotion planning problem for virtual characters evolving in a dynamic environment using a novel technique. While the legs and the pelvis of the virtual character follows a planned path, the animation of the upper part of the body is updated for 3D collision avoidance purposes (this happens normally in absence of dynamic obstacles, see Figure 5). In the presence of dynamic obstacles in its path, the reactive locomotion control takes the control and generates commands to stop the character and to let pass the intruder obstacles before forcing its DVZ to the original state.

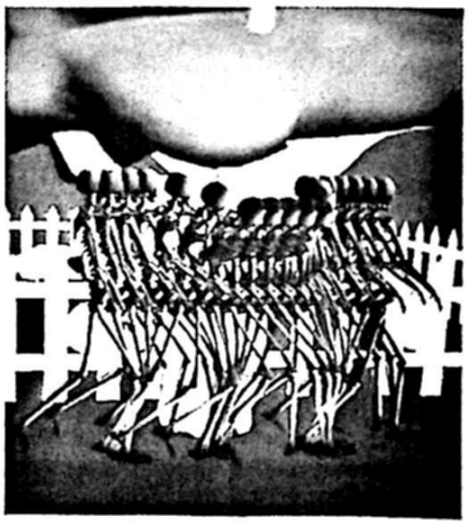


Fig. 5. Avoiding a branch.

Let us remember how it is possible to solve the collision avoidance problem for the reactive degrees of the character by using the warping module. The objective of the warping module is to locally modify the animation of the upper bodies of the character (arms and spine) when collisions occur in the animation produced by the locomotion controller. Each key-frame of the sequence is scanned and a collision test is performed. If a collision exists, the frame is marked. All the marked frames are gathered into connected subsequences, which are extended to create blocks absorbing collision-free frames in the neighborhood of the colliding subsequences. Such a subsequence extension is considered to provide smooth motions able to anticipate the corrective actions to be done. Each connected frame block is then processed independently. By following a similar idea to this proposal, it is possible to evaluate the deformation of the DVZ in each frame. Figure 6 illustrates our approach.

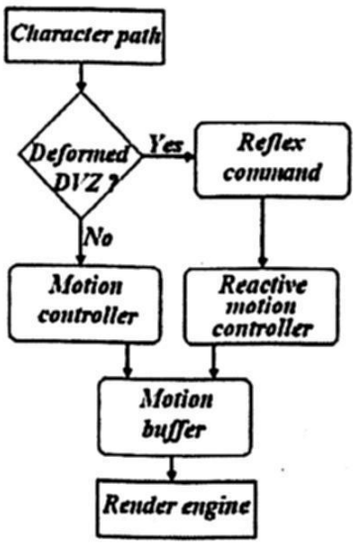


Fig. 6. An architecture for animation in dynamic environments.

In this work, we used a model of character of 52 degrees of freedom. The motion library is provided by the CMU Graphics Lab Motion Capture Database. Figure 7 illustrates the execution of the planner in a complex environment, a mall. The figures were trimmed, to show the part where the character walks and finds dynamic obstacles. The reflex commands in our approach are simples, because when the character finds dynamic obstacles, the most obvious and natural form to react is stopping the action (walk or running).

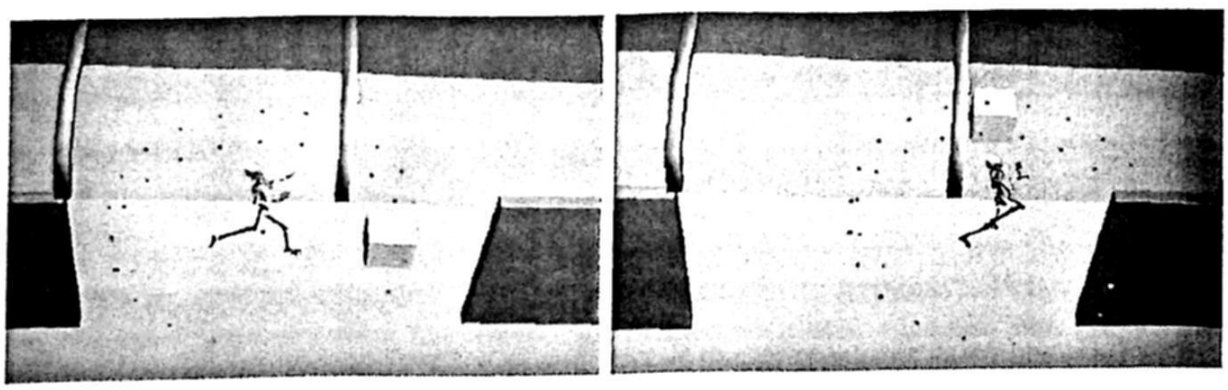


Fig. 7. Snapshots showing the action of the reactive controller to avoid dynamic obstacles.

The automated synthesis of motion for characters in unstructured environments is difficult because it requires solving a planning problem subject to multiple constraints. Obstacles in the environment constrain the motion in an obvious fashion, as typified by narrow passages. Other types of constraints include a character's joint limits, the requirements for balance, the character's natural disposition for particular postures and motion, and so on.

In this work, we have presented a novel idea to solve the locomotion planning problem for virtual characters in dynamic environments. This approach provides good results and mainly the running times are acceptable. Obviously, the free parameters of the DVZ are adjusted by hand, but they do not present greater problems. Also, we made tests in situations where the character must avoid dynamic obstacles and at the same time it must avoid collisions with static obstacles (when the upper part of the body encounters them) (see Figure 8).

5 Conclusions and Future Work

In recent years, robotics has been subject to promising advances in sensor and actuators hardware, sensory processing techniques and low-level control methods. Yet, the are has not been benefited to the full amount from the availability of powerful knowledge presentation tools and action calculi.

Many of the designers of reactive systems look to biology as a source of models for use in robots. Although the diversity of these efforts is significant, ranging from traditionally engineered systems to those that dedicate themselves

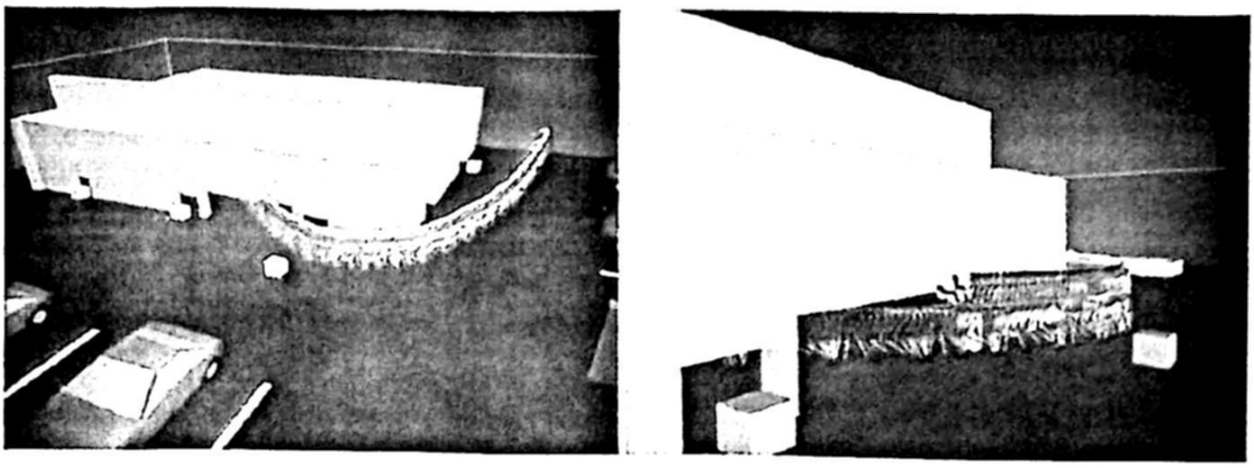


Fig. 8. Avoiding dynamic and static obstacles.

to faithfully replication biological behavior, this work reports on two examples that have affected reactive and hybrid system design.

The algorithms implementing the DVZ method can be seen as low-level interactions with the robot environment and must be coupled with high-level motion planning procedures. We have implicitly assumed that these high levels of control were existing to let the robot come back to its initial mission after a purely "reactive behavior".

A reactive lazy PRM planner for dynamically changing environments is presented in this paper. The results obtained in the evaluation of the reactive lazy PRM planner proposed in this work, show the importance of finding a solution for the complex problem of motion planning in dynamic environments.

Although some promising results are shown in its present form, the planner can be improved in a number of important ways. This approach can be extended to use real robots and to solve the problem posed by small static obstacles. Besides, some cases where the reflex actions are not sufficient to avoid collisions, were observed during the evaluation tests. Theses cases are difficult because they require a more intelligent behavior in order to avoid the robot to be trapped. In those cases, it can be necessary to add a process that computes the trajectories of moving objects and corrects the robot's path in real time.

Interactive generation of reactive motions for virtual humans as they are hit, pushed and pulled are very important to many applications, such as computer games, movies, cartoons, virtual environments. The use of motion captured data has become a popular mean of animating virtual characters, but since it is an off-line process, there has been great interest in developing algorithms that are suitable for interactive applications.

In this work, we have presented a novel idea to solve the locomotion planning problem for virtual characters in dynamic environments. This approach provides good results and mainly the running times are acceptable. Obviously, the free parameters of the DVZ are adjusted by hand, but they do not present greater problems. Also, we made tests in situations where the character must avoid

dynamic obstacles and at the same time it must avoid collisions with static obstacles (when the upper part of the body encounters them).

The DVZ method can be used obviously with vision (instead of using other sensors like the telemetric or laser). For the collision avoidance scheme, the main input is the distance field in the robot front space. Stereovision allows this 3D reconstruction by measuring the disparity field in two images. By capturing images at successive step times, two or more cameras can also be used to detect motion in the robot workspace. Assuming that almost all points in the robot workspace are static, it is also possible to derive an expression of the ego-motion (self motion of the robot) and to fuse it with proprioceptive information.

References

1. Bohlin R. and Kavraki L. E. "Path planning using lazy PRM", *Proc. of the IEEE Robotics and Automation Conference*, (2000) 521-528
2. A. Cacitti, R. Zapata. Reactive behaviours of mobile manipulator based on the DVZ method, *Proc. of the IEEE Int. Conf. on Robotics and Automation*, (2001) 680-685
3. Kavraki L. E., Švetska P., Latombe J. C., and Overmars M. H. Probabilistic roadmaps for path planning in high-dimensional configuration spaces, *IEEE Transactions on Robotics and Automation*, Vol. 12, No. 4, (1996) 566-580
4. O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots, *International Journal of Robotic Research*, Vol. 5, No. 1, (1986) 90-98
5. T. Komura, H. Leung and J. Kuffner. Animating reactive motions for biped locomotion, *ACM VRST 04*, (2004)
6. Lanzoni C., Sánchez L. A., and Zapata, R. "A single-query motion planner for car-like nonholonomic mobile robots", *3rd International Symposium on Robotics and Automation (ISRA)*, (2002) 267-274
7. J. C. Latombe. Robot motion planning, *Kluwer Academic Publishers*(1991)
8. A. Sánchez L., J. Sánchez T. and R. Zapata, Planning motions for animated characters, *Journal Research on Computing Science*, Vo. 17, (2005) 265-274
9. J. Sánchez T., A. Sánchez L., M. Rodriguez C., X. Hernández V. and G. Garcia L., RMP3D: A multipurpose platform for motion planning, *Journal Research in Computing Science*, Vol. 20, (2006) 167-178
10. A. Sánchez L., R. Cuautle P., M. Osorio L. and R. Zapata. A reactive lazy PRM approach for non-holonomic motion planning, *J. S. Sichman et al. (Eds.): IBERAMIA-SBIA 2006, LNAI 4140*, (2006) 542-551
11. R. Zapata, P. Lépinay and P. Thompson. Reactive behaviors of fast mobile robots, *Journal of Robotic Systems*, Vol. 11, No. 1, (1994) 13-20
12. V. Zordan and J. Hodgins. Motion capture-driven simulations that hit and react, *ACM Computer Animation*, (2002)

Haptic Guided Exploration of Deformable Objects

Omar Arturo Domínguez Ramírez[1] and Vicente Parra Vega[2]

[1] Universidad Autónoma del Estado de Hidalgo
Centro de Investigación en Tecnologías de Información y Sistemas
Carretera Pachuca-Tulancingo Km 4.5, Pachuca, Hidalgo, México
[2] Robotics and Advanced Manufacturing Division-CINVESTAV
Carretera Saltillo-Monterrey Km 13.5 Ramos Arizpe, Coahuila 25900, México.
omar@uaeh.reduaeh.mx, omar@uaeh.edu.mx, vparra@cinvestav.mx

Abstract. Perception and interaction with virtual objects through kinesthetic sensation and visual stimuli is the basic issue of a haptic interface. If a real object is located at a remote station and explored (in contact) with a passive device, a haptic interface in a local station can be used to perceive its spatial and surface attributes. This is one type of haptic guidance. This problem has been addressed with undeformable object, and contact force modelled with the penalty-based method. However, this approach yields limited haptic properties of the object, and if the object is deformable, it is difficult to achieve stable contact. However, there exists relevant tasks for exploration of deformable objects, such as exploration of fruits, skin of animals and dermatological procedures. Motivated by these kind of tasks, an approach for guided remote exploration of deformable objects is proposed in this paper. A real object is explored in a remote location and object attributes and properties such as spatial location, shape, texture and roughness are perceived with a constrained Lagrangian-based decentralized force-position controller in the local station. Stable interaction is theoretical proved and experimental results using PHANToM 1.0A validate the approach.

1 Introduction

1.1 Haptic Interface

The kinesthetic perception is possible by means of the use of an electromechanical device (haptic device) in closed loop with the virtual object [1]. The high bandwidth of the tactile force-pressure physiological sensor of the operator requires high precision haptic interface to stimulate correctly the mechanoreceptors of the operator, while low bandwidth haptic rendering is required for visual stimuli. However, the haptic rendering graphics should convey deformation of the virtual world accordingly to the bilateral force-pressure stimulus [1]. In particular, the PHANToM haptic interface [1], [2] has successfully being used for this purpose, though its application programming interface GHOST has limited capabilities since only simple undeformable primitives can be programmed. Therefore, simple

© S. Torres, I. López, H. Calvo. (Eds.)
Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 243-254

Received 24/02/07
Accepted 08/04/07
Final version 23/04/07

spring-based contact force models can be implemented. Though recent GHOST version allows simple dynamic properties to the virtual environments, and Phantom uses simple PID-based cartesian stiffness control, nowadays Phantom stands for a high end haptic interface, however this two facts further limits the scope of this haptic device for dynamic-based virtual environments, this device may become a powerful haptic interface if stable interaction with deformable objects can be established. On the other hand, some application for haptic teleoperation or haptic guidance have been proposed. In particular, new haptic guidance schemes are under research, wherein a master passive robot guides the remote haptic robot under different configurations.

1.2 Haptic Training

Haptic guidance can be used for training[12]. We identify four classes haptic guidance: **Configuration 1.** *Haptic Guidance:* The master sent only its position and contact force as desired references to the remote haptic interface. **Configuration 2.** *Haptic Guidance Control:* As configuration 1, but the master also controls the remote haptic interface. The difference between 1 and 2 is that in configuration 2 the master station controls directly the remote station, while configuration 1 implements an independent control loop in the remote station. **Configuration 3.** *Haptic Guided Exploration:* As definition 1, but the master performs a recognition task. **Configuration 4.** *Haptic Guided Exploration Control:* As definition 3, but the master controls the position and contact force of the remote haptic interface. The difference between 3 and 4 is that in configuration 4 the master station controls directly the remote station, while configuration 3 implements an independent control in the remote station. Finally, the difference between guidance and guided exploration is that a guided exploration configuration involves perception of shape, texture, and roughness, in contrast to a guidance configuration wherein this object attributes does not play a significant role. For instance, in some guided exploration tasks back and forth, and lateral motions might be important, while in guidance this movements are not important to complete the task. In this paper, we are interested in *Haptic Guidance Exploration* (configuration 3), which is useful for rapid training.

2 The Problem and A Solution

2.1 The Problem

How can haptic exploration of a deformable object be performed when an expert is training an inexpert? This interaction involves two haptic interfaces and two human operators, therefore compliant interaction arises. This new paradigm in haptic interfaces has been poorly explored, and [12] offers an extraordinary review on this subject. Haptic exploration involves contact to deformable object. This deformation may come from the deformation of the object itself due to the contact force, or, if the object is very stiff, deformation may occur due to the

compliance of whole system. Remember that the human operator is driving the system, thus a compliant contact action occurs with deformation. It is evident that the solution of this problems requires contact force based on the dynamics of the whole system. From this viewpoint, it is also evident that Hooke´s Law-based contact force will not suffice since a static mapping is used. It was shown in our previous paper [9], [11] that that interaction with the constrained lagrangian allows stable interaction in contrast to the contact force model based on the Hooke´s Law (called penalty-based method, which is hugely popular among the haptic research community due to the simplicity, despite of its limited performance and prone to instability). The high-end kinesthetic coupling that arises using the constrained Lagrangian method yields a more realistic contact force as a function on the dynamical properties of the whole system. An illustrative example of haptic guided exploration could be an expert surgeon carrying out a surgical exploration (the master is in contact to a real object), and in a remote station a inexpert surgeon is holding a haptic display for training purposes. However, this scheme allows control to the inexpert surgeon to test its abilities. Or an expert painter training an candidate painter while the expert painter is painting. For this two cases, it is relevant not to implement direct control from the master to the remote station. The master can be real [13] or virtual.

2.2 The Solution

A haptic guidance scheme for guided exploration is implemented to yield an active haptic exploration with purposes of remote training, with simultaneous control of force and position on the remote station. Second order Lagrangian dynamics are assigned to the virtual objects, and to the PHANToM haptic device. Then, constrained Lagrangian algorithm is implemented to compute the reaction force based on the dynamic properties of the dynamical virtual world and the PHANToM device. The components of this contact force is used to reproduce object attributes such as shape, texture and roughness to allow a more realist contact force compliant to the real sensations of remote exploration.

3 Exploration of a Remote Real Object

There are two robotic systems involved in haptic guidance. In the remote station, the remote robot can be a passive linkage robotic arm exploring the real object in contact; this remote arm is equipped with angular position sensors and force sensor to measure angular displacement and real contact forces with the object. At the local station, a haptic display is required to generate the force contact coming from the real contact at the remote station. In this paper, we consider that two Phantoms are involved in each side.

3.1 Constraint Dynamics of PHANToM at Each Side

PHANToM 1.0A is a mechanism of articulate links, with n revolute joints described in generalized joint coordinates $(q^T, \dot{q}^T)^T \in R^{2n}$. The dynamics presents

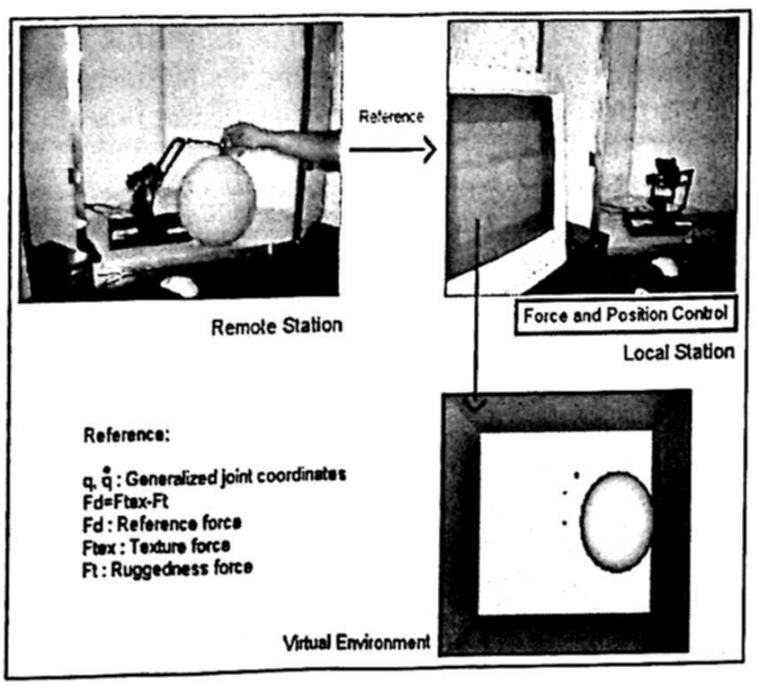


Fig. 1. Remote exploration of a real object through a haptic interface with haptic guidance controller

restriction in its movement, defined by the following algebraic and differential system of equations,

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau + \frac{J_\varphi^T}{\|J_\varphi J_\varphi^T\|} f_r \tag{1}$$

$$\varphi(q) = 0 \tag{2}$$

where $M(q) \in \Re^{3\times3}$ denotes a symmetric positive definite inertial matrix, $C(q,\dot{q}) \in \Re^{3\times3}$ is a Coriolis and centripetal forces matrix, $g(q) \in \Re^n$ models the gravity forces, $\tau \in \Re^3$ stands for the torque input, $f_r \in \Re^r$ is (for r=1 is one point contact, a scalar) constrained Lagrangian representing the magnitude of the contact force, $\frac{J_\varphi^T}{\|J_\varphi J_\varphi^T\|}$ stands for the normalized projection of the jacobian $J_\varphi \in \Re^n$, $J_\varphi = J_\varphi(q) \equiv \left[\frac{\partial}{\partial q_1}\varphi(q)\ \frac{\partial}{\partial q_2}\varphi(q)\ \frac{\partial}{\partial q_3}\varphi(q)\right]$, which arises normal at the contact point. The following equation holds while the end-effector is moving on the constraint surface $\varphi(q) = 0$,

$$\dot{\varphi}(q) = J_\varphi \dot{q} \equiv 0 \tag{3}$$

and,

$$\ddot{\varphi}(q) = J_\varphi \ddot{q} + \dot{J}_\varphi \dot{q} \equiv 0 \tag{4}$$

These equations must be satisfied for consistency of the solution of the DAE system.

3.2 The Local Station

Real Remote Object The surface of the object is described by a geometric function $\varphi(q) = 0$. Based in the constrained dynamic model (1)-(2). The real object can be modelled in terms of the generalized coordinates q since $\varphi(q) = 0$ as a mass-spring-damper system as follows

$$m\ddot{\xi}(q) + b\dot{\xi}(q) + k\xi(q) = 0 \tag{5}$$

where m is the mass, b is damper and k is spring. This pointwise model is consistent to the formulation of one point contact of the DAE system (1)-(2).

Computation of Contact Force for Local Phantom It is assumed that there exists a force sensor that delivers f_r in the remote station.

3.3 The Remote Station

Virtual remote object Similar to subsection 3.2, where now the virtual object is assigned a lumped second order linear dynamics with respect to its inertial frame.

Computation of Contact Force for Local Phantom Phantom is not equipped with a force sensor. Then, we propose to compute it by solving the DAE system (1)-(2) for f_r as follows. First for stable interaction ($\varphi(q) = 0$), the haptic display must stay in contact to the virtual object, then the acceleration $\xi(q)$ must equal the acceleration $\varphi(q)$, that is $\xi(q) = \varphi(q)$, and then (5) becomes

$$m\ddot{\varphi}(q) + b\dot{\varphi}(q) + k\varphi(q) = 0 \tag{6}$$

Using (4), equation (6) becomes

$$m(J_\varphi \ddot{q} + \dot{J}_\varphi \dot{q}) + b\dot{\varphi}(q) + k\varphi(q) = 0 \tag{7}$$

Solving (6) by using (1) we obtain

$$\ddot{q} = M(q)^{-1} \left\{ -C(q,\dot{q})\,\dot{q} - G(q) + \tau + \frac{J_\varphi^T}{\|J_\varphi J_\varphi^T\|} f_r \right\} \tag{8}$$

Now, compute the constrained Lagrangian f_r from (7) and (8) as follows

$$f_r = \frac{\|J_\varphi J_\varphi^T\|}{mJ_\varphi M(q)^{-1} J_\varphi^T} \left\{ -b\dot{\varphi}(q) - k\varphi(q) - m\dot{J}_\varphi \dot{q} + mJ_\varphi M(q)^{-1}(C(q,\dot{q})\,\dot{q} + G(q) - \tau) \right\} \tag{9}$$

Notice that the constrained Lagrangian f_r is function of J_φ, and dynamics of the haptic device and the object.

3.4 Reproducing Object Properties in the Remote Station

Equation (9) represents the reaction force in terms of: **i)** the PHANToM dynamics; **ii)** the dynamics of the virtual object, and **iii)** the controller τ. Notice that acceleration is not required. In this way, the controller τ will track (reproduce) the desired trajectories, that is the real contact force $f_{r-local}$ of the local station becomes the desired contact force for the haptic device in the remote station, that is $f_{rd-remote} = f_{r-local}$, and thus $q_{d-remote} = q_{local}$ for position. Since the object in the remote station exhibit roughness and texture through the contact force f_{rd}, then if a controller τ guarantees that $f_{r-remote}$ converges to $f_{rd-remote}$, then it also guarantees that the real object properties are perceived in the remote station. Now, since there exists at least a three degree of freedom force sensor in the local station, then $f_{r-local} = [f_x, f_y, f_z]^T$, and in the next section we propose how to parameterize roughness, shape and texture in terms of $[f_x, f_y, f_z]$ and object parameters. Notice that these properties are parameterized by the operational contact forces at each unitary axis i, j, k, which are available from the force sensor and friction parameters. In this way, since τ generate tracking of $[f_x, f_y, f_z]$, it will guarantee also tracking of roughness, shape and texture.

4 Computation of Texture, Roughness, and Haptic Exploration

How to reproduce object properties with only force sensor measurement when sliding over a real remote object? In this section, we discuss an approach that synthesizes texture, roughness, and shape from f_x, f_y, and f_z measurements.

4.1 Roughness Perception

The sliding friction between two different materials with contact area defined by A, is equal to the load W divided by the flow stress P_m of the weaker of the two solids in contact. At this region of contact, the solid form a number of junctions as if they were welded together. Friction F represents the force required to shear these junctions apart. Mathematically, the theory is expressed as,

$$A = \frac{W}{P_m} \tag{10}$$

$$F = As \tag{11}$$

$$\mu = \frac{F}{W} = \frac{s}{P_m} \tag{12}$$

where s is the shear stress. Thus, the coefficient of friction $\mu \ll 1$ may be represented by the ratio of shear stress to flow stress of the material, and becomes its intrinsic property. Roughness arises as function of the sliding motion over the surface of the object, thus roughness is function of the tangential friction f_T.

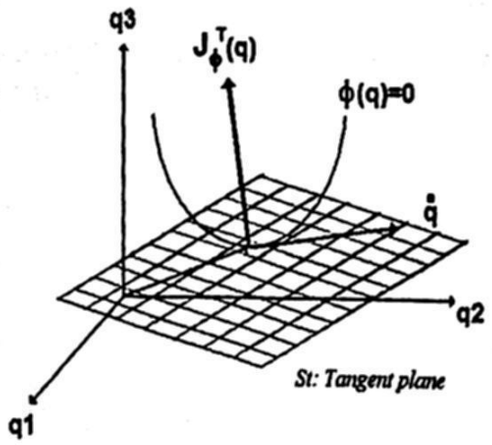


Fig. 2. Geometric decomposition in the contact point

Since f_T arises at the tangential plane at the contact point, see figure 2, it is then function of joint velocity.$\dot{q}$, in terms of $\dot{X} = J\dot{q}$ as follows

$$f_T = \mu\sqrt{(f_x^2 + f_z^2)}\dot{X}$$

where, $f_x, f_z \in S_t$. $\dot{X} = J\dot{q}$, with J as the Phantom jacobian matrix. The torques based on the tangential friction force is defined by the equation,

$$\begin{aligned} \tau_t &= J^T f_T \\ \tau_t &= \mu\sqrt{(f_x^2 + f_z^2)}J^T J\dot{q} \end{aligned} \tag{13}$$

In this way, we can model the roughness by simply assigning values to f_x, f_z, or these variables can be generated on line by the master station.

4.2 Texture Perception

The perception of surface texture is a specific design issue in force feedback interfaces. Manipulation of everyday objects, the perception of surface texture is fundamental to accurate identify contact points and apply the correct internal contact force. In a virtual environment also, haptic texture information can both increase the sense of realism of an object as well as convey information about what the object is and where it is. Phantom haptic device convey texture by actuating kinesthetic forces on the users fingers. In this work we model the texture property as a periodic function

$$T_{tex} = Amp\,(\sin(2\pi f t) + 1) \tag{14}$$

where Amp stands for half of the maximum value of texture torque, f stand for the frequency in hertz and t is the time in seconds.

4.3 Shape Perception

Shape is perceived by the normal contact force of an object. Thus, equation (9) directly provides this perception in absence of roughness and texture.

5 Haptic Exploration Control

Haptic guidance schemes are employed in tasks of remote training. The haptic device defines, in the station teacher, the position references (free motion) or position and forces (constrained motion) that will reproduce in the remote device (station remote). Experiment is shown in diagram of the figure 1. We use a nonlinear PID control [4], [7] for free movement experiments, and a simultaneous control of force and position for constrained movement experiments with the human on the loop [7], [8],[10].

5.1 Free Motion Control of the Remote Station

A nonlinear PID control [4] is proposal for haptic guidance task in free movement. This control compensate the nonlinear dynamics in continuous mechanical plants with tracking capability. The nonlinear PID controller given by

$$\tau = -k_p \Delta q - k_v \Delta \dot{q} + k_d S_d - k_i \int_{t_0}^{t} sgn(\Delta \dot{q} + \alpha \Delta q - S_d) d\varsigma \tag{15}$$

where $S_d = (\Delta \dot{q}(t_0) + \alpha \Delta q(t_0)) \exp^{-k(t-t_0)}$, for $k_p, k_v, k, \alpha > 0$, are positive feedback gains of appropriate dimensions. Tracking errors are defined as $\Delta q = q - q_d, \Delta \dot{q} = \dot{q} - \dot{q}_d$ for position and velocity, respectively. Desired values are the real position and velocity of the local station. This controller guarantees exponential tracking without using the model, see [4].

5.2 Constrained Motion Control of the Remote Station

We makes use of our previously proposed control law [7]

$$\tau = -\tau_p - \tau_f - \tau_d \tag{16}$$

where

$$\begin{aligned} \tau_p &= -K_p(t)\Delta q - K_v(t)\Delta \dot{q} - K_{ip}(t) I_p \\ \tau_p &= K_F(t)\Delta F + K_\lambda(t)\Delta \lambda - K_{iF}(t) I_f \\ \tau_d &= K_g \mathcal{N} + \zeta(t) \end{aligned} \tag{17}$$

and $K_p(t), K_v(t), K_{ip}(t), K_{iF}(t), K_F(t), K_\lambda(t), \dot{I}_F, \dot{I}_p, K_g(t), \mathcal{N}, \zeta(t)$ are time varying feedback gains that depends on matrix $Q(q) = I - J_\varphi^T(q)\left(J_\varphi(q) J_\varphi^T(q)\right)^{-1}$ stands for the orthogonal projection of the normal of a matrix $J_\varphi \in \Re^{1x3}$, and on J_φ^T. Gains α, β are positive constants, $\Delta F = \int (f_r - f_{rd}) dt$, $K_d = K_d^T \in \Re^{nxn}$, f_d are positive gains. This controller guarantees fast simultaneous tracking of position and force trajectories defined by the master operator. See [7].

5.3 A Stable Switching Algorithm

Remote exploration involves free and constrained motion, that at least two controllers are switching over time (it was shown that switching of these controllers is stable). The algorithm is as follows

- Phase a (*Without interaction*): $\varphi(q) > \varepsilon \rightarrow$ Free Motion Control
- Phase b (*Collision detection*): $-\varepsilon \leq \varphi(q) \leq \varepsilon \rightarrow$ Constrained Motion Control
- Phase c (*Stable interaction with deformation*): $\varphi(q) < -\varepsilon \rightarrow$ Constrained Motion Control

where $\varepsilon = 1 \rightarrow 10^{-6}$ m. It can be seen that the applying the constrained Lagrangian method, in contrast to the penalty-based method, involves low frequencies over the virtual object. This allows a stable interaction, as proved in [14], without trembling for deformable objects.

6 Experiments

6.1 Experimental Setup

The conditions of the experiments are defined in a parallel plane S_t to the plane $X - Z$, the human operator of the remote station develops a circular trajectory on the plane S_t with texture and ruggedness (in way emulated by means of references to the controller of force and position in the local station). The constraint surface is $\varphi(q) = l_2 - l_2\cos(q_3) + l_1\sin(q_2) - y_0$, where $l_1 = l_2 = 139.7$ mm. The parametric equations that define the trajectory are, $x = h + r\cos wt, y = y_0, z = k + r\sin wt)$, this equations correspond to a circumference in the plane S_t, with center in $C(h, y_0, k)$ and radio r. The Jacobian of PHANToM is given by

$$J = \begin{bmatrix} l_1c_1c_2 + l_2s_3c_1 & -l_1s_1s_2 & l_2s_1c_3 \\ 0 & l_1c_2 & l_2s_3 \\ -(l_1s_1c_2 + l_2s_1s_3) & -l_1s_2c_1 & l_2c_1c_3 \end{bmatrix} \tag{18}$$

where $c_* = \cos(*)$ and $s_* = \sin(*)$. The time in all the experiments is of $t = 5$ seconds. In all the experiments, the following parameters were used, $h = -25.0$ $mm, y_0 = 20.0$ $mm, k = 0.0$ $mm, w = \frac{2\pi}{4} radsec^{-1}$. To define the texture the equation (14) is used, and to define f_x and f_z the following equation is implemented,

$$f_x = R\sin(2\pi f_R t)\left[\frac{J_{\varphi_{11}}}{\|J_\varphi J_\varphi^T\|}\right] \tag{19}$$

$$f_z = R\sin(2\pi f_R t)\left[\frac{J_{\varphi_{13}}}{\|J_\varphi J_\varphi^T\|}\right]$$

In the table 1 presents the parameters used in each one of the experiments,

Experiment	*Amp*	f	μ	R	f_R
1	0	0	0	0	0
2	150	0.5	0.015	50	0.5
3	150	1.5	0.015	50	1.5
$4 - deformation$	150	0.5	0.015	50	0.5

Table 1. Parameters used in experiments

6.2 Results

Fig. 3 describes the experiments of configuration 1, given in section 1. It can be seen that when there is not direct interaction of the master station over the local (inexpert) station, smooth haptic performance is obtained. Fig. 4 describes the experiments of configuration 2, given in section 1. Note that once texture and tangential friction is present over the path to be followed, the collision detection algorithm allows stable switching, while tracking is obtained. Fig. 5 describes the experiments of configuration 3, given in section 1. In this configuration local force-position control loops is implemented to allow kinesthetic stimuli of desired force-position trajectories, generated by the master. Texture does exhibit a frequency three fold of the previous case, and it is reproduced effortlessly with stable interaction. Fig. 6 describes the experiments of configuration 4, given in section 1. In this configuration direct control is performed through the master over the local station, wherein considerable deformation is achieved with stable contact, and even during deformation, texture and roughness is perceived seemingly.

7 Conclusions

A haptic system to allow remote exploration of a real object is proposed. A general framework based on constrained robot dynamics renders a Lagrangian-based contact force controller within a systematic way to produce shape, roughness and texture properties of the remote object under exploration. Even during deformation, these object properties can be perceived. The system is stable for free, collision and constrained motion by using a novel decentralized class of force-position robot control. Impedance can be easily incorporated, though it would not render tracking of both trajectories. This result has been supported theoretically, and experimental evidence suggest a successful exploration of dynamical (deformable) remote objects. Note that the lumped model of the object is calculated at each instant, and resembles the FEA since second order dynamics are computed in every point of the object. The computational cost is quite low, and yet formal stability arguments guarantee stability of the closed-loop system, critical for remote virtual training.

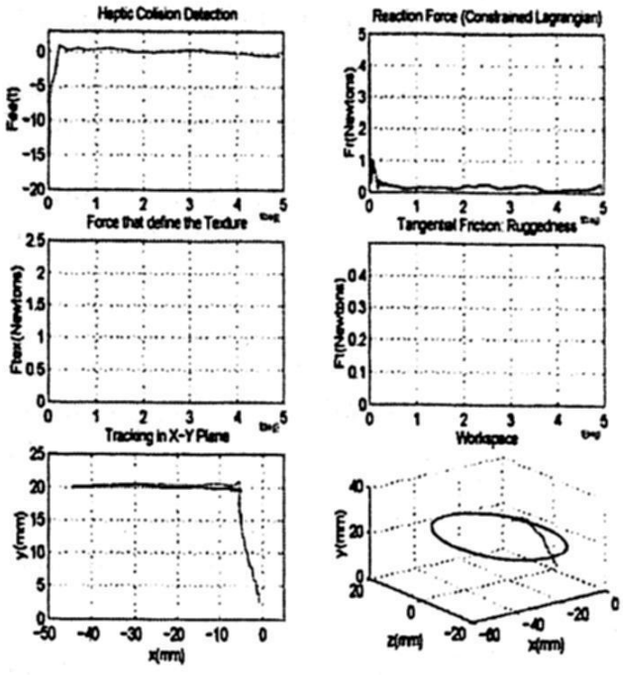


Fig 3. Experiment 1

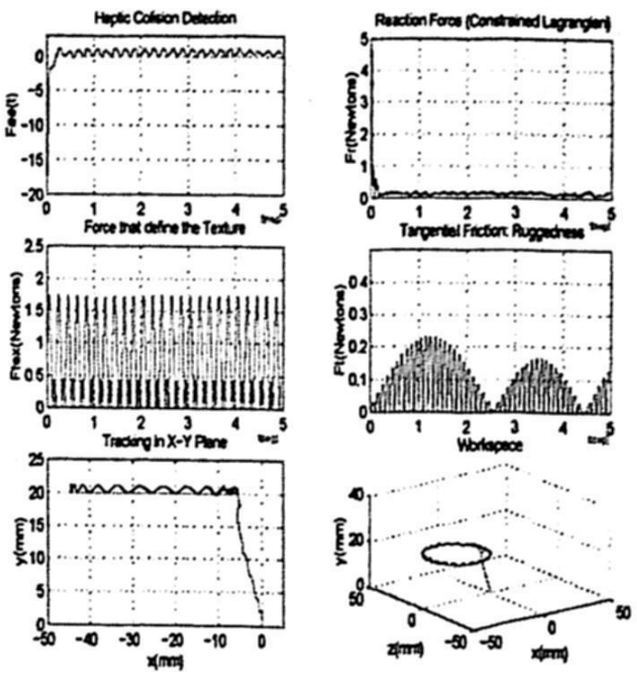


Fig 4. Experiment 2

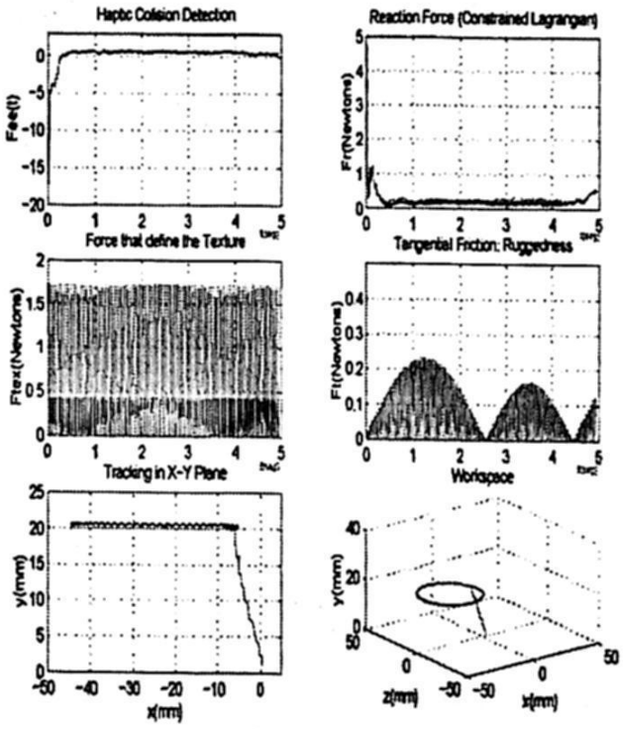


Fig 5. Experiment 3

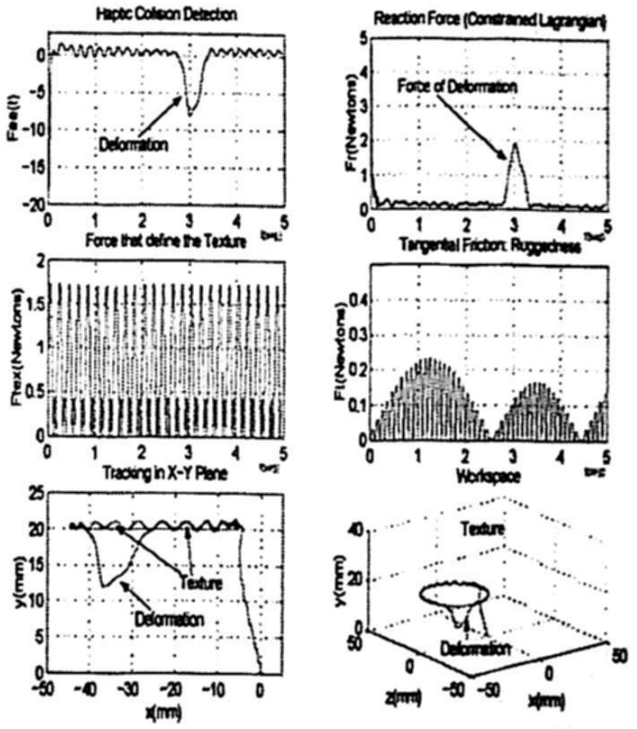


Fig 6. Experiment 4

References

1. G.C. Burdea, *"Force and Touch Feedback for Virtual Reality"*, John Wiley and Sons, Inc., Electrical and Computer Engineering Department, The State University of New Jersey, 1996.

2. Salisbury, J.K. and Mandayam A. Srinivasan. *"Phantom-Based Haptic Interaction with Virtual Objects"*, IEEE Computer Graphics and Applications; Volume 17, Number 5, pp. 6-10; September-October 1997.
3. M. Moreyra, B. Hannaford, *"A Practical Measure of Dynamic Response of Haptic Devices"*, International Conference on Robotics and Automation, Proceedings of the 1998 IEEE, pp. 369-374, Leuven Belgium, 1998.
4. V. Parra-Vega, S. Arimoto *"Nonlinear PID Control with Sliding Modes for Tracking of Robot Manipulator"*, Conference on Control Applications. Proceedings of the 2001 IEEE International.
5. C. Preusche, G. Hirzinger, *"Scaling Issues for Teleoperation"*, The Fifth Phantom user's Group Workshop, Massachusetts Institute of Technology, Cambridge, MA, pp. 39-42, 2000.
6. C.R. Carignan, K.R. Cleary, *"Closed-Loop Force Control for Haptic Simulation of Virtual Environments"*, The Electronic Journal of Haptics Research, vol. 1, No. 2, 2000.
7. V. Parra-Vega, A. Castillo-Tapia, *"Perfect Tracking of Force and Position for Constrained Robot Manipulators with Orthogonal Nonlinear PID Control"*, International Symposium on Robotics and Automation 03, Toluca, México 2003.
8. V. Parra-Vega,L.G. García-Valdovinos, A. Castillo-Tapia and Omar A. Domínguez-Ramírez, *"Sliding PID Control for Tracking in Finite Time for Robot Arms"*, The 11th International Conference on Advanced Robotics, pp. 1526-1531, University of Coimbra, Coimbra, Portugal, 2003.
9. Omar A. Domínguez-Ramírez and Vicente Parra-Vega, *"Constrained Lagrangian-Based Force-Position Control Haptic Guidance"*, pp. 444-450, EUROHAPTICS 2003, Trinity College Dublin and Media Lab Europe, Dublin, Ireland, 2003.
10. Omar A. Domínguez-Ramírez and Vicente Parra-Vega, *"Active Haptic Interface with Purposes of Remote Training"*, The 11th International Conference on Advanced Robotics, pp. 1588-1593, University of Coimbra, Coimbra, Portugal, 2003.
11. O. A. Dominguez-Ramirez, V. Parra-Vega, *"Texture, Roughness, and Shape Haptic Perception of Deformable Virtual Objects with Constrained Lagrangian Formulation"*, IIEEE-RSJ International Conference on Intelligent Robotts and Systems, October 27 - 31, 2003 Bally"s Las Vegas Hottell, Las Vegas,, NV, USA
12. D. Feygin, M. Keehner, and F. Tendick, *"Haptic Guidance: Experimental Evaluation of a Haptic Training Method for a Perceptual Motor Skill"*, Proceedings 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, part of IEEE Virtual Reality 2002, Orlando, FL, pp. 40–47, Mar. 2002
13. A. Okamura, M.R. Kutkosky, and J.T. Dennerlein, *"Reality-based Models for Vibration Feedback in Virtual Environments"*, IEEE/ASME Trnasactions of Mechatronics, Vol. 6, No. 3, september 2001, pp. 245-252.
14. P. Akella, V. Parra-Vega., Arimoto S. and T. Tanie, *"Discontinuous Adaptive Control for Robot Manipulators Executing Free and Contrained Tasks"*, Proc. of the IEEE Robotics and Automation, San Diego (1994), pages 3000-3007.

An Intelligent Traffic Signal Control Based on EVALPSN

Kazumi Nakamatsu[1] and Jair Minoro Abe[2]

[1] University of Hyogo, Himeji, JAPAN
nakamatu@shse.u-hyogo.ac.jp
[2] Paulista University, Sao Paulo, BRAZIL
jairabe@uol.com.br

Abstract. We introduce two kinds of traffic signal control methods based on a paraconsistent annotated logic program called EVALPSN. One is for single traffic signal control and another one is for coordinated traffic signal control. We explain both of the methods with a simple example of intersections and introduce the simulation results of the two kinds of EVALPSN traffic signal control methods by comparing them with two conventional ones.
Keywords : EVALPSN(Extended Vector Annotated Logic Program with Strong Negation), traffic signal control, paraconsistent annotated logic program, defeasible deontic reasoning.

1 Introduction

We have proposed a paraconsistent annotated logic program called EVALPSN (Extended Vector Annotated Logic Program) that can deal with defeasible deontic reasoning and paraconsistency [9, 10]. Some applications of EVALPSN to various kinds of control and safety verification, robot action control, discrete event control, and safety verification for railway interlocking and air traffic control have been introduced [9, 12, 15]. Moreover, we have shown that some specific EVALPSNs can be easily implemented on microchips [13].

Traffic jams caused by inappropriate traffic signal control are serious environmental issues that we have to overcome. Conventionally, traffic signal control is divided into three kinds; single, coordinated and broad area ones according to the number of traffic signals to be controlled. In single traffic signal control, each traffic light is controlled independently. On the other hand, in coordinated ones, a series of traffic lights on the same route are controlled.

Recently various intelligent traffic signal control methods in which fuzzy logic, neural networks, GA(Genetic Algorithm), etc. are used for optimizing control have been proposed [1, 5, 6, 16]. Furthermore a traffic signal control method based on GA has been applied in a real scenario [4, 19, 20]. However, the method takes a long time to compute optimal solutions and is not so appropriate for real-time control. We have already proposed an intelligent real-time single traffic signal control system based on EVALPSN and have shown that it could reduce the traffic density 10% to 15% in simulation [14]. In this paper, we extend the idea

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 255-266

Received 20/02/07
Accepted 08/04/07
Final version 16/04/07

of EVALPSN single traffic signal control to a coordinated one and propose an EVALPSN based coordinated traffic signal control system.

This paper is organized in the following manner: firstly, we review EVALPSN briefly; next we introduce the idea of EVALPSN single traffic signal control with a simple example, how to formalize it in defeasible deontic formulas, and how to translate them into EVALPSN; subsequently, we formalize the coordinated traffic signal control for the example in defeasible deontic formulas, and translate them in EVALPSN similarly; last, we show the simulation results of the EVALPSN based coordinated traffic signal control system by comparing them with two conventional ones.

We assume that the reader is familiar with the basic knowledge of logic programming[7].

2 EVALPSN

First we will briefly review EVALPSN, the details of EVALPSN are found in [9–11]. Generally, a truth value called an *annotation* is explicitly attached to each literal in annotated logic programs [2]. For example, let p be a literal, μ an annotation, then $p:\mu$ is called an *annotated literal.* The set of annotations constitutes a complete lattice. An annotation in EVALPSN has a form of $[(i,j),\mu]$ called an *extended vector annotation.* The first component (i,j) is called a *vector annotation* and the set of vector annotations constitutes the complete lattice,

$$\mathcal{T}_v(n) = \{\ (x,y) | 0 \leq x \leq n, 0 \leq y \leq n, x, y \text{ and } n \text{ are integers}\ \}$$

in **Fig.1.** The ordering($\preceq_v$) of the lattice $\mathcal{T}_v(n)$ is defined as : let $(x_1,y_1),(x_2,y_2) \in \mathcal{T}_v(n)$,

$$(x_1,y_1) \preceq_v (x_2,y_2) \textbf{ iff } x_1 \leq x_2 \text{ and } y_1 \leq y_2.$$

For each extended vector annotated literal $p:[(i,j),\mu]$, the integer i denotes the amount of positive information to support the literal p and the integer j denotes that of negative one. The second component μ is an index of fact and deontic notions such as obligation, and the set of the second components constitutes the complete lattice,

$$\mathcal{T}_d = \{\bot, \alpha, \beta, \gamma, *_1, *_2, *_3, \top\}.$$

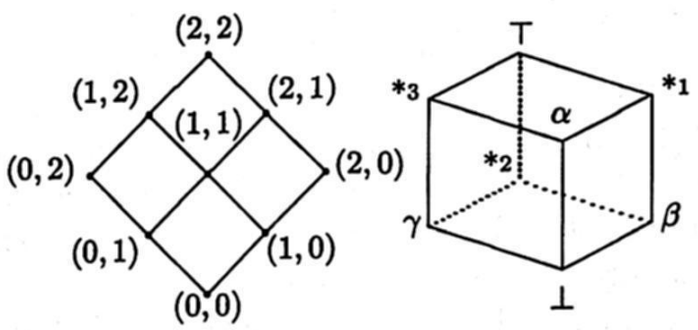


Fig. 1. Lattice $\mathcal{T}_v(2)$ and Lattice $\mathcal{T}_d$

The ordering($\preceq_d$) of the lattice $\mathcal{T}_d$ is described by the Hasse's diagram in **Fig.1.** The intuitive meaning of each member of the lattice $\mathcal{T}_d$ is $\perp$ (unknown), α (fact), β (obligation), γ (non-obligation), $*_1$ (fact and obligation), $*_2$ (obligation and non-obligation), $*_3$ (fact and non-obligation), and $\top$ (inconsistency). Then the complete lattice $\mathcal{T}_e$ of extended vector annotations is defined as the product $\mathcal{T}_v(n) \times \mathcal{T}_d$. The ordering($\preceq_e$) of the lattice $\mathcal{T}_e$ is defined as follows :
let $[(i_1, j_1), \mu_1]$ and $[(i_2, j_2), \mu_2] \in \mathcal{T}_e$,

$$[(i_1, j_1), \mu_1] \preceq_e [(i_2, j_2), \mu_2] \quad \textbf{iff} \quad (i_1, j_1) \preceq_v (i_2, j_2) \text{ and } \mu_1 \preceq_d \mu_2.$$

There are two kinds of *epistemic negation* ($\neg_1$ and $\neg_2$) in EVALPSN, which are defined as mappings over $\mathcal{T}_v(n)$ and $\mathcal{T}_d$, respectively.

Definition 1(epistemic negations $\neg_1$ and $\neg_2$ in EVALPSN)

$$\begin{aligned}
&\neg_1([(i,j),\mu]) = [(j,i),\mu], \quad \forall \mu \in \mathcal{T}_d \\
&\neg_2([(i,j),\perp]) = [(i,j),\perp], \quad \neg_2([(i,j),\alpha]) = [(i,j),\alpha], \\
&\neg_2([(i,j),\beta]) = [(i,j),\gamma], \quad \neg_2([(i,j),\gamma]) = [(i,j),\beta], \\
&\neg_2([(i,j),*_1]) = [(i,j),*_3], \quad \neg_2([(i,j),*_2]) = [(i,j),*_2], \\
&\neg_2([(i,j),*_3]) = [(i,j),*_1], \quad \neg_2([(i,j),\top]) = [(i,j),\top].
\end{aligned}$$

If we regard the epistemic negations as syntactical operations, the epistemic negations followed by literals can be eliminated by the syntactical operations. For example, $\neg_1 p{:}[(2,0),\alpha] = p{:}[(0,2),\alpha]$ and $\neg_2 q{:}[(1,0),\beta] = p{:}[(1,0),\gamma]$.

There is another negation called *strong negation* ($\sim$) in EVALPSN, and it is treated as classical negation.

Definition 2(strong negation $\sim$ in EVALSPN)
Let F be any formula and $\neg$ be $\neg_1$ or $\neg_2$.

$$\sim F =_{def} F \rightarrow ((F \rightarrow F) \wedge \neg(F \rightarrow F)).$$

Definition 3 (well extended vector annotated literal)
Let p be a literal. $p : [(i, 0), \mu]$ and $p : [(0, j), \mu]$ are called *weva(well extended vector annotated)-literals*, where $i, j \in \{1, 2, \cdots, n\}$, and $\mu \in \{ \alpha, \beta, \gamma \}$.

Defintion 4 (EVALPSN)
If $L_0, \cdots, L_n$ are weva-literals,

$$L_1 \wedge \cdots \wedge L_i \wedge \sim L_{i+1} \wedge \cdots \wedge \sim L_n \rightarrow L_0$$

is called an *EVALPSN clause*. An *EVALPSN* is a finite set of EVALPSN clauses.

Fact and deontic notions, "obligation", "forbiddance" and "permission" are represented by extended vector annotations, $[(m, 0), \alpha]$, $[(m, 0), \beta]$, $[(0, m), \beta]$, and $[(0, m), \gamma]$, respectively, where m is a positive integer. For example,

- $p{:}[(2,0),\alpha]$ is intuitively interpreted as "it is true of strength 2 that p is a fact";
- $p{:}[(1,0),\beta]$ is as "it is true of strength 1 that p is obligatory";
- $p{:}[(0,2),\beta]$ is as "it is false of strength 2 that p is obligatory", that is to say, "it is true of strength 3 that p is forbidden";
- $p{:}[(0,1),\gamma]$ is as "it is false of strength 1 that p is not obligatory", that is to say, "it is true of strength 1 that p is permitted".

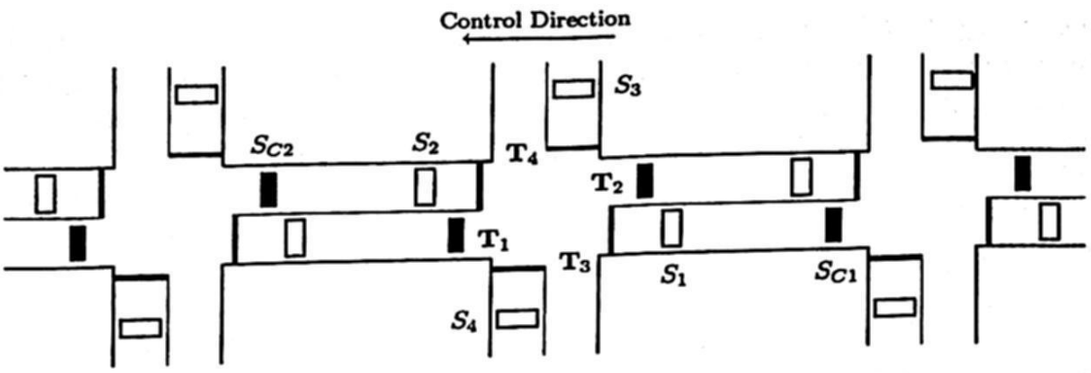


Fig. 2. Intersections in Japan

3 Traffic Signal Control in EVALPSN

First of all we suppose a basic traffic signal control policy in which traffic lights treating a larger amount of traffic should have the priority to be green until reducing traffic density. Based on this control policy the traffic signal is controlled in real-time by defeasible deontic reasoning in EVALPSN.

Suppose that you are waiting for the traffic light turning from red to green at an intersection. Then, you must have a demand for turning the traffic light from red to green in your mind. On the other hand, if you are driving through the intersection with green you must have a demand for keeping the traffic light green. The first demand can be regarded as permission for turning the light from red to green and the second one can be regarded as forbiddance from turning the traffic light from green to yellow or red. Then, there is a conflict between the permission and forbiddance. The basic idea of EVALPSN traffic signal control is to resolve such conflicts by EVALPSN defeasible deontic reasoning. We will formalize the traffic signal control in defeasible deontic formulas firstly and translate them into EVALPSN later.

3.1 Single Traffic Signal Control

We take a series of three typical intersections in Japan as described in **Fig.2**. In single traffic signal control, we focus on only the middle intersection among the three in **Fig.2** as the object of the single traffic signal control. We assume that:

- the middle intersection has four traffic lights $T_{1,2,3,4}$, each one has three kinds of displays, green, yellow and red, and they have the following signal chart,

$$\begin{aligned} T_{1,2} &\rightarrow \text{red} \rightarrow \text{red} \rightarrow \text{all red} \rightarrow \text{green} \rightarrow \text{yellow} \rightarrow \text{all red} \rightarrow \\ T_{3,4} &\rightarrow \text{green} \rightarrow \text{yellow} \rightarrow \text{all red} \rightarrow \text{red} \rightarrow \text{red} \rightarrow \text{all red} \rightarrow \end{aligned}$$

- the intersection has four sensors $S_{1,2,3,4}$ to detect traffic density or flowing traffic, which are described as white boxes in **Fig.2**;
- yellow and all red terms are fixed and only green term is variably controlled according to the traffic detected by the sensors;

Table 1. Defeasible Reasoning

Superiority Relation	a	b	conclusion
R1 < **R2**	true	true	$\neg p$
R1 < **R2**	true	false	p
R1 < **R2**	false	true	$\neg p$
No relation	true	true	nothing
No relation	true	false	p
No relation	false	true	$\neg p$

- the minimum and maximum lengths of green term are given in advance, that is to say, green term must be controlled between the minimum and maximum terms.

For example, if the sensor S_1 detects flowing traffic, the green term of the traffic light T_1 should be extended within the range between the minimum and maximum terms. In the EVALPSN traffic signal control, the flowing traffic or traffic density of each road connected to the intersection are regarded as the forbiddance or permission for turning the object traffic light from green to yellow, and the conflict between the permission and forbiddance should be resolved by EVALPSN defeasible deontic reasoning. As the first step, we will formalize the traffic signal control in defeasible deontic formulas and translate them into EVALPSN.

Defeasible Deontic Rules for Single Traffic Signal Control Firstly, we introduce defeasible reasoning briefly. The details are found in [17, 18]. Basically, two kinds of rules, a strict rule $A \rightarrow p$ and a defeasible rule $B \Rightarrow q$ are used in defeasible logic [17], where A,B and p,q are called the prerequisites and consequents of the rules $A \rightarrow p$ and $B \Rightarrow q$, respectively. Defeasible reasoning is carried out based on the superiority relations between defeasible rules. Strict rules are used for deriving facts and superior to defeasible rules. Suppose that there are conflicting defeasible rules :

$$\textbf{R1} \quad a \Rightarrow p \qquad \text{and} \qquad \textbf{R2} \quad b \Rightarrow \neg p.$$

Then, we have the defeasible reasoning results in **Tab.1**. For example, if we take the first line. it shows that: if there is the superiority relation **R1** < **R2**, and both a and b are true, then the rule **R2** defeats the rule **R1** and only the consequent $\neg p$ of the defeasible rule **R2** is derived.

Now we will start formalizing the single traffic signal control for the middle intersection in defeasible deontic formulas. We suppose that the traffic lights $T_{1,2}$ and $T_{3,4}$ are red and green, respectively. Then, the following conditions should be considered as strict or defeasible rules.

1. If one of the sensors $S_{1,2}$ detects traffic density more than the criterion at the time t, the permission for turning the traffic lights $T_{3,4}$ from green to yellow

should be derived. This condition is represented by the defeasible rule,

$$S^{rg}_{1,2}(t) \wedge T_{1,2}(r,t) \wedge T_{3,4}(g,t) \Rightarrow \neg \bigcirc \neg T_{3,4}(y,t), \quad (1)$$

where, $S^{rg}_{1,2}(t)$ denotes that one of the sensors $S_{1,2}$ detects the traffic density over the criterion at the time t; $T_{1,2}(r,t)$ and $T_{3,4}(g,t)$ denote that the traffic lights $T_{1,2}$ and $T_{3,4}$ are red and green at the time t, respectively; and the symbol $\bigcirc$ is a modal operator to denote obligation, thus, the symbols $\neg\bigcirc\neg$ and $\bigcirc\neg$ denote permission and forbiddance, respectively.

2. On the other hand, if one of the sensors $S_{3,4}$ detects the flowing traffic over the criterion at the time t, the forbiddance from turning the traffic lights $T_{3,4}$ from green to yellow should be derived. Then, we have the defeasible rule conflicting with the defeasible rule (1),

$$S^{rg}_{3,4}(t) \wedge T_{1,2}(r,t) \wedge T_{3,4}(g,t) \Rightarrow \bigcirc \neg T_{3,4}(y,t). \quad (2)$$

We have to define the superiority relation ($<$) between the defeasible rules (1) and (2). As we assume that maintaining the current signal state is superior to turning it to the next signal state, the forbiddance from turning the traffic lights $T_{3,4}$ from green to yellow is prior to the permission for it, that is to say, the defeasible rule (2) is considered to be superior to the defeasible rule (1), i.e.,(1) $<$ (2).

We need more traffic signal control rules. If we take only the defeasible rules (1) and (2) into account, we might have an extreme situation such that the green term of $T_{1,2}$ is 2 hours but that of $T_{3,4}$ is 1 minute,. In order to avoid such an extreme control, we have to control green term in an appropriate range, which are represented by the following definite rules.

Minimum Green Term Rule Each green term of the traffic lights $T_{1,2,3,4}$ should be guaranteed its minimum length. Let $MIN_i(g,t)$, $(i \in \{1,2,3,4\})$ denote that the green time of the traffic light T_i has not passed its minimum term at the time t. Then we have a rule : if the green time is shorter than its minimum term at the time t, it is forbidden from the traffic light turning from green to yellow. This rule can be represented by the strict rules,

$$MIN_{1,2}(g,t) \wedge T_{1,2}(g,t) \rightarrow \bigcirc \neg T_{1,2}(y,t), \quad (3)$$
$$MIN_{3,4}(g,t) \wedge T_{3,4}(g,t) \rightarrow \bigcirc \neg T_{3,4}(y,t). \quad (4)$$

Maximum Green Term Rule Each green term of the traffic lights $T_{1,2,3,4}$ also should have its maximum term. Let $MAX_i(g,t)$, $(i \in \{1,2,3,4\})$ denotes that the green time of the traffic light T_i has already passed its maximum term at the time t. Then we have a rule : if the green time is longer than its maximum term at the time t, it is definitely permitted for the traffic light turning from green to yellow. This rule can be represented by the strict rules,

$$MAX_{1,2}(g,t) \wedge T_{1,2}(g,t) \rightarrow \neg \bigcirc \neg T_{1,2}(y,t), \quad (5)$$
$$MAX_{3,4}(g,t) \wedge T_{3,4}(g,t) \rightarrow \neg \bigcirc \neg T_{3,4}(y,t). \quad (6)$$

There are conflicting defeasible rules (1) and (2) whose consequents are permission and forbiddance. As a defeasible reasoning result, if the permission is derived at the time t, the traffic light has to be turned from green to yellow at the next time $t+1$. On the other hand, if the forbiddance is defeasibly derived at the time t, the signal has to keep the current state at the next time $t+1$. These conditions can be represented by the following strict rules :

$$T_{1,2}(g,t) \wedge \neg \bigcirc \neg T_{1,2}(y,t) \rightarrow \bigcirc T_{1,2}(y,t+1), \tag{7}$$
$$T_{3,4}(g,t) \wedge \neg \bigcirc \neg T_{3,4}(y,t) \rightarrow \bigcirc T_{3,4}(y,t+1), \tag{8}$$
$$T_{1,2}(g,t) \wedge \bigcirc \neg T_{1,2}(y,t) \rightarrow \bigcirc T_{1,2}(g,t+1), \tag{9}$$
$$T_{3,4}(g,t) \wedge \bigcirc \neg T_{3,4}(y,t) \rightarrow \bigcirc T_{3,4}(g,t+1). \tag{10}$$

Moreover, we need more strict rules for synchronizing all the traffic lights. For example, if it is obligatory that the traffic light T_1 is red at the time t, the traffic light T_2 also must be red at the same time, and if it is obligatory that the traffic light T_3 is green at the time t, the traffic light T_1 must be red at the same time. Such synchronization is represented by the following strict rules,

$$\bigcirc T_1(r,t) \rightarrow \bigcirc T_2(r,t) \quad \text{and} \quad \bigcirc T_3(g,t) \rightarrow \bigcirc T_1(r,t) \tag{11}$$

EVALPSN for Single Traffic Signal Control Now, we will translate the strict and defeasible rules into EVALPSN. The strict and defeasible rules can be translated by formalizing their semantics in EVALPSN. The details of the translation are described in [8]. The defeasible rules (1) and (2) are translated into the EVALPSN clauses,

$$\begin{aligned} &{S^{rg}}_{1,2}(t):[(2,0),\alpha] \wedge \\ &T_{1,2}(r,t):[(2,0),\alpha] \wedge T_{3,4}(g,t):[(2,0),\alpha] \wedge \\ &\sim MIN_{3,4}(g,t):[(2,0),\alpha] \wedge \\ &\sim {S^{rg}}_3(t):[(2,0),\alpha] \wedge \sim {S^{rg}}_4(t):[(2,0),\alpha] \\ &\rightarrow T_{3,4}(y,t):[(0,1),\gamma], \end{aligned} \tag{12}$$

$$\begin{aligned} &{S^{rg}}_{3,4}(t):[(2,0),\alpha] \wedge \\ &T_{1,2}(r,t):[(2,0),\alpha] \wedge T_{3,4}(g,t):[(2,0),\alpha] \wedge \\ &\sim MAX_{3,4}(g,t):[(2,0),\alpha] \rightarrow T_{3,4}(y,t):[(0,1),\beta]. \end{aligned} \tag{13}$$

The EVALPSN clause (12) represents that :

if the traffic sensors S_1 or S_2 have detected the traffic density over the criterion, the minimum green term has already passed, and
the traffic sensors neither S_3 nor S_4 has detected the flowing traffic over the criteria at the time t under the condition that the traffic lights $T_{3,4}$ are green,
then the traffic lights $T_{3,4}$ are permitted for turning to yellow.

On the other hand, the EVALPSN clause (13) represents that :

if the traffic sensors S_3 or S_4 have detected the flowing traffic over the criterion, and
the maximum green term has not passed yet under the same condition that the traffic lights $T_{3,4}$ are green,
then the traffic lights $T_{3,4}$ are forbidden for turning to yellow.

Similarly the strict rules (4), (6), (8) and (10), are also translated into the EVALP clauses,

$$MIN_{3,4}(g,t):[(2,0),\alpha] \wedge T_{3,4}(g,t):[(2,0),\alpha] \rightarrow T_{3,4}(y,t):[(0,2),\beta], \quad (14)$$
$$MAX_{3,4}(g,t):[(2,0),\alpha] \wedge T_{3,4}(g,t):[(2,0),\alpha] \rightarrow T_{3,4}(y,t):[(0,2),\gamma], \quad (15)$$
$$T_{3,4}(g,t):[(2,0),\alpha] \wedge T_{3,4}(y,t):[(0,1),\gamma] \rightarrow T_{3,4}(y,t+1):[(2,0),\beta], \quad (16)$$
$$T_{3,4}(g,t):[(2,0),\alpha] \wedge T_{3,4}(y,t):[(0,1),\beta] \rightarrow T_{3,4}(g,t+1):[(2,0),\beta]. \quad (17)$$

Example 1
Suppose that the traffic lights $T_{1,2}$ are red and the traffic lights $T_{3,4}$ are green, furthermore, it has already passed the minimum green term then.

Case-1. If one of the sensors $S_{1,2}$ detects the traffic density over the criterion at the time t and the rest of the sensors do not react then, the EVALPSN clause (12) is fired and the permission $T_{3,4}(y,t):[(0,1),\gamma]$ is derived, furthermore, the EVALPSN clause (16) is also fired and the obligation $T_{3,4}(y,t+1):[(2,0),\beta]$ is also derived.

Case-2. If both the sensors S_1 and S_3 detect the traffic density and flowing traffic over the criteria at the time t, respectively, and the rest of the sensors do not detect then, the EVALPSN clause (13) is fired and the forbiddance $T_{3,4}(y,t):[(0,1),\beta]$ is derived, furthermore, the EVALPSN clause (17) is also fired and the obligation $T_{3,4}(g,t+1):[(2,0),\beta]$ is also derived.

3.2 Coordinated Traffic Signal Control

We exhibit the basic idea of the EVALPSN coordinated traffic signal control by taking the same intersections in **Fig.2** as an example. Here we focus on only the traffic from the right to the left and its converse direction as the object of the EVALPSN coordinated traffic signal control. In conventional coodinated traffic signal control, not only each term of green, yellow and red but also a timelag called an *off set* between two neighbor traffic lights are controlled, and they should be pre-installed. On the other hand, in EVALPSN coordinated traffic signal control, only the green term of each coordinated traffic light should be controlled by EVALPSN defeasible deontic reasoning based on traffic amount information, which is detected by traffic sensors attached to not only the object intersection but also its both of neighbor intersections. Therefore, when we consider the EVALPSN coordinated traffic signal control in terms of the middle intersection, the extra traffic sensors $Sc_{1,2}$ described by black boxes in **Fig.2** attached to the right and left intersections should be taken into account to detect the traffic flowing into the middle intersection. Moreover, we note that when the

traffic information detected by the remote sensors $Sc_{1,2}$ is processed, appropriate timelags between the detection and processing of the traffic information should be taken into account according to the distances between the object intersection and the sensors $Sc_{1,2}$.

Defeasible Deontic Rules for Coordinated Traffic Signal Control We assume the same condition as the single traffic signal control, that is to say the traffic lights $T_{1,2}$ and $T_{3,4}$ are red and green, respectively. Then, we have the following conditions in defeasible deontic formulas.

3. If one of the sensors Sc_1 or Sc_2 detects flowing traffic over the criteria at the time t, the permission for turning the traffic lights $T_{3,4}$ from green to yellow should be derived. Then, we have the defeasible rule,

$$Sc^{rg}_{1,2}(t) \wedge T_{1,2}(r,t) \wedge T_{3,4}(g,t) \Rightarrow \neg \bigcirc \neg T_{3,4}(y,t), \tag{18}$$

where $Sc^{rg}_{1,2}(t)$ denotes that one of the sensors $Sc_{1,2}$ detects the flowing traffic over the criterion.

We also need to consider the superiority relation between the conflicting defeasible rules (2) and (18). As the coordinately controlled route traffic from the right to the left or its converse are regarded to be superior to other traffic, the permission for turning the traffic lights $T_{3,4}$ from green to yellow has a prior to the forbiddance from it, that is to say, the defeasible rule (18) is considered to be superior to the defeasible rule (2), i.e.,(2) $<$ (18).

EVALPSN for Coordinated Traffic Signal Control Taking the superiority relations among the defeasible rules (1), (2) and (18), i.e., (1) $<$ (2) $<$ (18) into account, those defeasible rules are retranslated into the EVALPSN clauses,

$$\begin{aligned}
&S^{rg}{}_{1,2}(t):[(2,0),\alpha] \wedge \\
&T_{1,2}(r,t):[(2,0),\alpha] \wedge T_{3,4}(g,t):[(2,0),\alpha] \wedge \\
&\sim MIN_{3,4}(g,t):[(2,0),\alpha] \wedge \\
&\sim S^{rg}{}_3(t):[(2,0),\alpha] \wedge \sim S^{rg}{}_4(t):[(2,0),\alpha] \\
&\rightarrow T_{3,4}(y,t):[(0,1),\gamma],
\end{aligned} \tag{19}$$

$$\begin{aligned}
&S^{rg}{}_{3,4}(t):[(2,0),\alpha] \wedge \\
&T_{1,2}(r,t):[(2,0),\alpha] \wedge T_{3,4}(g,t):[(2,0),\alpha] \wedge \\
&\sim MAX_{3,4}(g,t):[(2,0),\alpha] \wedge \\
&\sim Sc^{rg}{}_1(t):[(2,0),\alpha] \wedge \sim Sc^{rg}{}_2(t):[(2,0),\alpha] \\
&\rightarrow T_{3,4}(y,t):[(0,1),\beta],
\end{aligned} \tag{20}$$

$$\begin{aligned}
&Sc^{rg}{}_{1,2}(t):[(2,0),\alpha] \wedge \\
&T_{1,2}(r,t):[(2,0),\alpha] \wedge T_{3,4}(g,t):[(2,0),\alpha] \wedge \\
&\sim MIN_{3,4}(g,t):[(2,0),\alpha] \rightarrow T_{3,4}(y,t):[(0,1),\gamma],
\end{aligned} \tag{21}$$

As well as the EVALPSN single traffic signal control the coordinated one also has **Minimum** and **Maximum Green Term Rules**, therefore, we also have the EVALPSN clauses (14), (15), (16), (17) as the translation of those rules.

Example 2

Suppose the same conditions as **Example 1.**

Case-3. If one of the sensors $Sc_{1,2}$ and the sensor S_4 detect the flowing traffic over the criteria at the time t, respectively, the EVALPSN clause (21) is fired and the permission $T_{3,4}(y,t) : [(0,1),\gamma]$ is derived, furthermore, the EVALPSN clause (16) is also fired and the obligation $T_{3,4}(y,t+1) : [(2,0),\beta]$ is also derived.

Case-4. If all the sensors $S_{1,2,3,4}$ and $Sc_{1,2}$ detect the flowing traffic or traffic density over the criteria at the time t, respectively, the EVALPSN clause (21) is fired and the permission $T_{3,4}(y,t) : [(0,1),\gamma]$ is derived, furthermore, the EVALPSN clause (16) is also fired and the obligation $T_{3,4}(y,t+1) : [(2,0),\beta]$ is also derived.

Case-5. If the sensors $S_{1,2}$ and $Sc_{1,2}$ detect the traffic density and flowing traffic over the criteria at the time t, respectively, both the EVALPSN clauses (19) and (21) are fired and the same permission $T_{3,4}(y,t) : [(0,1),\gamma]$ is derived, furthermore, the EVALPSN clause (16) is also fired and the obligation $T_{3,4}(y,t+1) : [(2,0),\beta]$ is also derived.

4 Simulation

We now present the simulation results of the four kinds of traffic signal control, conventional (fixed time) single, coordinated, EVALPSN single, and EVALPSN coordinated ones for the intersections in **Fig.2** by the cellular automaton simulation method. We have assumed the following conditions for simulation,

- the unit time called "step" is defined in the simulation system, which is the time that a car travels one cell to the next cell;
- 5000 steps/hour;
- each distance between the three intersections is the same 20 cells;
- cars are flowing into all the intersections from each road in 10% probability except for the traffic from the right road, which is 15%, where 15% probability means 15 cars/100 steps appear at the road;
- for fixed-time traffic signal control, the green term is 60 steps, the yellow term is 3 steps, the off-set in the coordinated traffic signal control is 24 steps;
- for EVALPSN traffic signal control, the green term is between 14(min) and 60(max) steps, the yellow term is fixed 5 steps.

We took the simulation data of 50 cars that were chosen at random among all the cars flowing into the intersections from the right and traveling to the left. All four kinds of simulation have been carried out during 5000 steps each and repeated ten times. The simulation data in **Tab.2** show the average values of the ten times simulation results, where

"total car" denotes the total number of cars that flowed into all the three intersections from any road ;
"stop step" denotes the total number of steps that the 50 cars stopped ;
"travel step" denotes the total number of steps for the 50 cars having traveled from the right to the left ;
"travel car" denotes the total number of cars that traveled from the right up to the left.

Table 2. Simulation Results

	total car	stop step	travel step	travel car
Fixed-time Single	3804	66	177	495
Fixed-time Coordinated	3823	65	173	500
EVALPSN Single	4149	43	152	577
EVALPSN Coordinated	4169	34	122	607

This simulation results clearly say that : EVALPSN traffic signal control shows more efficient results than a conventional fixed time one in both single and coordinated traffic signal control methods, especially EVALPSN coordinated control is most efficient among them.

5 Conclusion and Future Work

We have introduced two kinds of traffic signal control methods for single and coordinated traffic signal controls as an application of EVALPSN with their computer simulation results by the cellular automaton method. Then we obtained the following conclusion : EVALPSN traffic control is a sensor based real-time one and more efficient than conventional ones according to the simulation results ; however, if it is implemented practically, lots of sensor installation are required and too much cost; EVALPSN control can be implemented in both existent software and hardware such as PLC(Programmable Logic Controller), although it has not been addressed in this paper; since each traffic light is controlled independently as one autonomous agent in the EVALPSN coordinated traffic signal control, it is more fault tolerant than conventional coordinated one.

We are planning to apply the EVALPSN traffic signal control methods to network control, for example, network routing control, electric power supply network control etc. in our future work.

References

1. Abdulhai,B.: Reinforcement Learning for the True Adaptive Traffic Signal Control. J.Transportation Engineering **129** (2003) 278–285
2. Blair,H.A., Subrahmanian,V.S.: Paraconsistent Logic Programming. Theoretical Computer Science **68** (1989) 135–154

3. Da Costa,N.C.A., Subrahmanian,V.S., Vago,C.: The Paraconsistent Logics P$\mathcal{T}$. Zeitschrift für Mathematische Logic und Grundlangen der Mathematik **37** (1989) 139–148
4. Foy,M.D., Benekohal,R.F., Goldberg,D.E.: Signal Timing Determination Using Genetic Algorithms. Transportation Research Record **1365** (1993) 108–105
5. Girianna,M., Benekohal,R.F.: Application of Genetic Algorithms to Generate Optimum Signal Coordination for Congested Networks. Proc. 7th Int'l Conf. Applications of Advanced Technology in Transportation, ASCE (2002) 762–769
6. Li,R., Li,J., Lu,H.: Multi-Layer Traffic Signal Control Model Based on Fuzzy Control and Genetic Algorithm. In Proc. 9th Int'l Conf. Applications of Advanced Technology in Transportation, ASCE (2006) 461–466
7. Lloyd,J.: *Foundations of Logic Programming*(2nd Edition). Springer-Verlag (1987)
8. Nakamatsu,K.: On the Relation Between Vector Annotated Logic Programs and Defeasible Theories. Logic and Logical Philosophy **8** (2001) 181–205
9. Nakamatsu,K.: Intelligent Information Systems Based on Paraconsistent Logic Programs. In Innovations in Intelligent Systems and Applications. Studies in Fuzziness and Soft Computing Series **140**, Springer-Verlag (2004) 257-283
10. Nakamatsu,K., Abe,J.M., Suzuki,A.: Annotated Semantics for Defeasible Deontic Reasoning. In Proc. the Second International Conference on Rough Sets and Current Trends in Computing, LNAI **2005**, Springer-Verlag (2001) 470–478
11. Nakamatsu,K., Abe,J.M., Suzuki,A.: A Defeasible Deontic Reasoning System Based on Annotated logic Programming. In Computing Anticipatory Systems CASYS2000, AIP Conference Proceedings **573**, AIP Press (2001) 467–478
12. Nakamatsu,K., Chung,S-L., Komaba,H., Suzuki,A.: A Discrete Event Control Based on EVALPSN Stable Model Computation. In Rough Set, Fuzzy Set, Data Mining and Granular Computing, LNAI **3641**, Springer-Verlag (2005) 671- 681
13. Nakamatsu,K., Mita,Y., Shibata,T.: An Intelligent Action Control System Based on Extended Vector Annotated Logic Program and its Hardware Implementation. Intelligent Automation and Soft Computing **13** (2007) 289–304
14. Nakamatsu,K., Seno,T., Abe,J.M., Suzuki,A.: Intelligent Real-time Traffic Signal Control Based on a Paraconsistent Logic Program EVALPSN. Rough Set, Fuzzy Set, Data Mining and Granular Computing, LNAI **2639** Springer-Verlag (2003) 719-723
15. Nakamatsu,K., Suito,H., Abe,J.M., Suzuki,A.: Paraconsistent Logic Program Based Safety Verification for Air Traffic Control. In Proc. 2002 IEEE Int'l Conf. Systems, Man and Cybernetics, IEEE (2002) CD-ROM
16. Nishikawa,I., Iritani,T., Sakakibara,K.: Improvement of the Traffic Signal Control by Complex-Valued Hopfield Networks. In Proc. Int'l Joint Conf. Neural Networks 2006, IEEE (2006) 1186–1191
17. Nute,D.: Basic Defeasible Logic. In Intensional Logics for Programming, Oxford Science Publications (1992) 126–154
18. Nute,D.: Apparent Obligation. In Defeasible Deontic Logic, Kluwer Academic Publishers (1997) 287–316
19. Park,B., Messer,C.J., Urbanik II,T.: Traffic Signal Optimization Program for Oversaturated Conditions: A Genetic Algorithm Approach. Transportation Research Record **1683** (1999) 133–142
20. Park,B., Messer,C.J., Urbanik II,T.: Enhanced Genetic Algorithms for Signal Timing Optimization of Oversaturated Intersections. Transportation Research Record **1727** (2000) 32–41

Fault Detection Combining PCA, Control Charts and Statistic Operation Limits

Juan Pablo Nieto González[1], Luis E. Garza Castañón[2]
and Rubén Morales Menendez [3]

[1] Mechatronics Department ITESM Saltillo campus
25000, Saltillo, Coah, Mexico
juan.pablo.nieto@itesm.mx
[2] Mechatronics and Automation Department ITESM Monterrey campus
64849, Monterrey, NL, Mexico
legarza@itesm.mx
[3] Center for Industrial Automation ITESM Monterrey campus
64849, Monterrey, NL, Mexico
rmm@itesm.mx

Abstract. As processes grow in size and complexity, the monitoring of them becomes more important, to avoid production losses and even accidents involving damage to personnel health and equipments. Process monitoring is particularly challenging due to the presence of both continuous and discrete variables, noisy information and lack or excess of data. There are two important tasks involved in the process monitoring: fault detection and fault diagnosis. This paper proposes a fault detection framework combining Principal Components Analysis (PCA), Control Charts and a comparison with Statistic Limits obtained from historical data process and inductive learning. PCA and control charts have been used in the past to detect suspicious observations. Once the suspicious observations are detected, a contribution chart and a comparison with the statistic limits are performed for fault detection. We show preliminary results from an electric circuit simulation composed by five subsystems.

Keywords: Fault Detection, Principal Component Analysis, Inductive Learning, Control Charts, Statistic Limits.

1 Introduction

As processes become more complex, the monitoring of them is very important in order to improve process performance, efficiency and product quality. Monitoring of industrial processes plays a substantial role in system safety, availability and production quality. Early detection of faults can help to avoid major breakdowns and incidents. In order to tackle those problems, fault detection and system diagnosis has been an active research domain since a few years ago.
There exist many research works related with fault detection. Most of the methods used are analytic, based on artificial intelligence (AI) or statistical methods.

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 267-278

Received 11/02/07
Accepted 08/04/07
Final version 22/04/07

[1] classifies fault detection and isolation methods in three groups. 1) Quantitative Model Based, 2) Qualitative Model Based and 3) Process History Based.
Quantitative Model Based fault detection methods are based on a mathematical model of the system. The occurrence of a fault is capture by discrepancies between the observed behavior and the prediction made by the model. These approaches use state estimation, parameter identification techniques, and parity relations to generate residuals. Fault localization then, rests on interlining the groups of components that are involved in each of the detected discrepancies. However, it is often difficult and a time-consuming task to develop accurate mathematical models that characterize all the physical phenomena occurring in industrial processes.
Qualitative Model based fault detection methods use symbolic reasoning which generally combines different sources of knowledge with graph theory to analyze the relationships between variables of a system. An advantage of these methods is that an explicit model of the system to be diagnosed is not necessary. Knowledge-based approaches such as expert systems, may be considered as alternative or complementary approaches where analytical models are not available.
Process History Based fault detection methods only require a big quantity of historical process data. There are several ways in which these data can be transformed and presented as prior knowledge of a system. These transformations are known as feature extraction and could be qualitative, as those used by expert systems, and qualitative trend analysis methods or quantitative, as those used in neural networks, PCA, PLS or statistical pattern recognition.
There are many papers related to fault detection and diagnosis in different processes. They use either an individual technique or make a combination of different techniques, taking advantage of the best characteristics of each method, to perform a better general behavior of the fault detection process. [2] presents a comparative study in the monitoring of hybrid systems, where the continuous part is modeled by Bond Graph and the discrete part is modeled by Petri Nets. [3] proposes a structure of a hybrid fault diagnosis system which integrates Signed Directed Graph, Artificial Neural Networks and dynamic simulation. [4] introduces a method based on hybrid system theory, which combines knowledge base methods and model base methods. [5] proposes a process monitoring which is composed of three parts: preanalysis, visualization and diagnosis, where the proposed method integrates PCA, FDA and clustering analysis taking advantage of each technique for a complete solution. [6] describes plant devices, sensors, actuators and diagnostic tests as stochastic finite state machines, by assigning transition probabilities and marginal probabilities to safe and fault events. By using simple composition rules, it is possible to determine the feasible configuration of alarms and their conditional probability given any event. [7] combines the use of signed directed graph to make a classification model, PCA and fuzzy knowledge to form a qualitative and quantitative model and compares the grade of the patterns needed to be diagnosed to the given fault patterns. [8] proposses a method based on the interaction between AI and control techniques. It uses a causal graph representation of the process, enabling decomposition into sub-

systems and reducing the diagnostic computational complexity. After that, at local level, FDI techniques based on numerical residual generation and analysis are carried out. [9] proposes a useful method when measures on the input signals can not be done due a nonexistent sensor or because it is impossible to do measurements due to the nature of the system itself. Thus it takes plant output signals, combines its variances, and uses a discriminant analysis upon the resultant features to carry out the diagnosis. In [10] PCA and statistical control charts are used to detect process operating faults on an industrial rolling mill reheating furnace. The Q statistic and Hotelling T^2 statistic are used to calculate the control limits of the statistical control chart. [11] proposes a fault diagnosis model based on machine learning which extracts multi-dimension features from the detected signal to supervise the different features of it simultaneously.
In this paper we propose a fault detection framework combining Principal Components Analysis (PCA), Control Charts and a comparison with Statistic Limits obtained from historical data process and inductive learning. We show preliminary results from an electric circuit simulation composed with five subsystems. The organization of the paper is as follows: section 2 gives preliminaries which explaines how the statistical limits are obtained as well as background knowledge on PCA and control charts. Section 3 gives the framework general description. Section 4 shows how the framework works in a simulation example with single and multiple faults and the performance of it in presence of white noise on measurements, as well as a comparison of the general performance of it against two similar frameworks. Section 5 gives conclusions to the paper.

2 Preliminary

2.1 Automatic Statistical Limits Obtention

[11] gives an algorithm to extract the statistical boundary vectors of a multidimensional feature extraction. In this paper a modification of that algorithm is done. Instead of doing multi-dimension feature extraction, here we work just with the statistical mean of the system variable being measured. Thus, the algorithm of inductive learning is used to obtain the statistical boundary vectors of $w_{max}(t_1, t_n, i)$ and $w_{min}(t_1, t_n, i)$ from a matrix which m_i rows are the different subsystems forming an entire process and the n columns present the changes of the statistical mean of the subsystem variable being measured as time changes from time t_1 to time t_n. The algorithm is shown below.

1. Initialize the statistical boundary vectors $w_{max}(t_1, t_n, i) = [w(t_1, t_2, 1) \cdots w(t_1, t_2, m_i)]^T$, $w_{min}(t_1, t_n, i) = [w(t_1, t_2, 1) \cdots w(t_1, t_2, m_i)]^T$ and the counter $j = 0$.
2. Calculate $w_{max}(i) = [max(w_{max}(t_1, t_n, 1), w_{max}(t_n, t_{n+1}, 1) \cdots max(w_{max}(t_1, t_n, m_i), w_{max}(t_n, t_{n+1}, m_i)))]^T$ and calculate $w_{min}(i) = [min(w_{min}(t_1, t_n, 1), w_{min}(t_n, t_{n+1}, 1) \cdots min(w_{min}(t_1, t_n, m_i), w_{min}(t_n, t_{n+1}, m_i)))]^T$. If $w_{max}(i) = w_{max}(t_i, t_n, i)$ and $w_{min}(i) = w_{min}(t_i, t_n, i)$, then $j = j + 1$, else $j = 0$.

3. If $j \geq V_1$, go to step(4), else $w_{max}(t_1, t_n, i) = w_{max}(i)$, $t_n = t_{n+1}$, go to step (2)
4. Output $w_{max}(t_1, t_n, i)$, $w_{min}(t_1, t_n, i)$, t_n, exit.

Where $w_{max}(t_1, t_n, i)$ and $w_{min}(t_1, t_n, i)$ in the present paper are used as the desired statistical limits for the statistical mean of the system variable being measured.

2.2 Principal Components Analysis (PCA)

The principal component analysis is concerned with explaining the variance and covariance structure of a set of variables, through a few linear combinations of these variables. The general objectives are basically: data reduction and interpretation.

PCA decomposes the X original data matrix with dimension $m \times n$ (m number of samples and n number of variables) as:

$$X = t_1 p_1^T + t_2 p_2^T + \cdots + t_n p_n^T + E = T_n P_n^T + E \tag{1}$$

t_i vectors are called the scores of the principal components and have information on how the samples are related to each other. p_i vectors are the eigenvectors of the covariance matrix of X, and are known as the loads of the principal components. They have information on how the variables are related to each other. In fact principal components analysis splits X matix in two parts, one that describes the system variation and other one that captures noise or information not modeled. The X matrix could often be aproximated using only A ($\leq n$) principal components instead of n variables as

$$\hat{X} = \sum_{i=1}^{A} t_i p_i^T + e \tag{2}$$

Where e is the residual. PCA is scale dependent, thus when variables are measured in different scales or on a common scale with widely differing ranges, they are often standardized. Another important issue is the minimum quantity of components needed to explain the data. The number of PC to retain in order to represent the maximum variance depends on the data and the existing correlation between the variables such that there are several decision criteria. [12] proposes to consider the amount of total sample variance explained, the relative sizes of eigenvalues or the use of scree plots. Thus the number of principal components should be equal or less than the variables of X. When the maximum variance of data is explained with the first two principal components, samples lie on a plane and a constant density ellipse could be formed by them.

Figure 1 shows the plane and constant density ellipse formed by two principal components, where the first principal component is the one that has the major data variation, while the second one is the next with the major data variation of the rest and is orthogonal to the first one. Thus, PCA model is able to describe significant variations in a fewer dimension than the original n variables does.

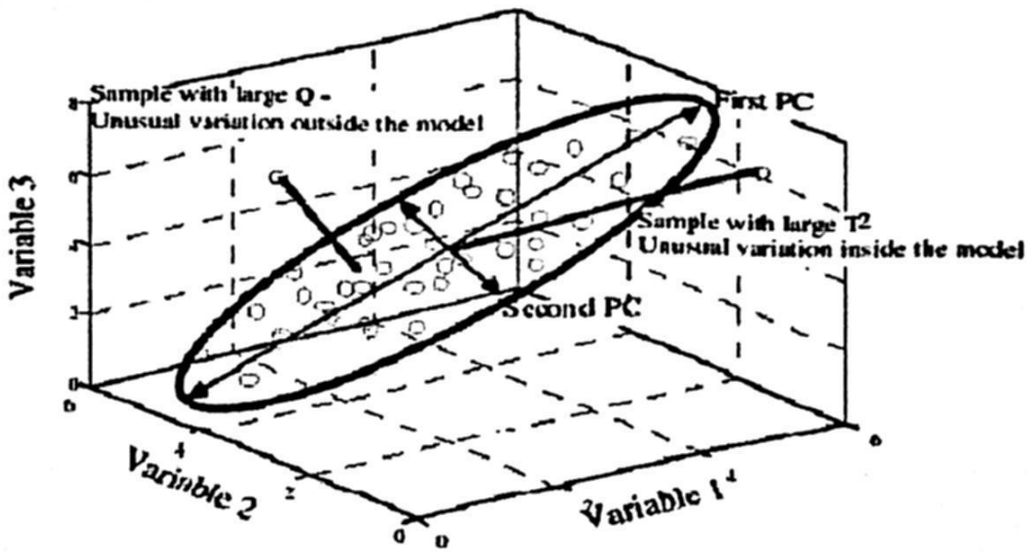


Fig. 1. Plane and constant density ellipse formed by two principal components.

2.3 Control and Contribution Charts in Statistical Process Control

Generally, there are two statistics to define the action and warnings limits used in multivariate control charts. The first statistic is Hotelling's T^2 as follows:

$$T_i^2 = X_i P \Lambda P^T X_i^T \sim \frac{A(n-1)}{n-A} F_{A,n-A} \tag{3}$$

Where X_i is the vector containing the data matrix X at sample time i, and Λ is a diagonal matrix containing the inverse of the eigenvalues of the PC scores. T^2 is a statistical measure of the multivariate distance of each observation from the center of the data set. This is an analytical way to find the most extreme points in the data. Thus, an out of control signal is identified if

$$T_i^2 > \frac{A(n-1)}{(n-A)} F_{A,n-A,\alpha} \tag{4}$$

confidence limit α typically takes the value of 0.05 or 0.01 for the limits. The second metric used in process monitoring to identify non-conforming operation is the Q statistic (also referred as Squared Prediction Error, SPE). The Q statistic is defined to be the quadratic form of the residuals, that is the squared difference between the observed values and predicted values from the nominal or reference models:

$$Q_i = e_i e_i^T = \sum_{j=1}^{k} (x_{ij} - \hat{x}_{ij})^2 \tag{5}$$

And its upper limit (UL) is given by a chi-square distribution with $p-A$ degrees of freedom

$$UL = \chi^2_{p-A}(\alpha) \tag{6}$$

Q is the statistic that measures lack of fit of a model to data. Under the assumption that the linear PCA is valid, the Q statistic defines the Euclidean

distance of the position of an observation from the hyperplane formed by the PCA model. See figure 1. In addition we could determine which variable of the process being analyzed is responsible for the unusual Q behavior, looking at a chart showing the contribution of each input to the Q statistic. This chart is known as the contribution chart and includes all process variables and their corresponding PCA scores in its axis.

3 Framework Description

The proposed detection framework is shown in figure 2. As the framework is a Process History Based fault detection method, this only requires a big quantity of historical data process. This data set takes into account only the normal system data operation. They will be transformed by both, PCA model and the normal operation data limits, and used as prior knowledge of the system to perform the detection process.

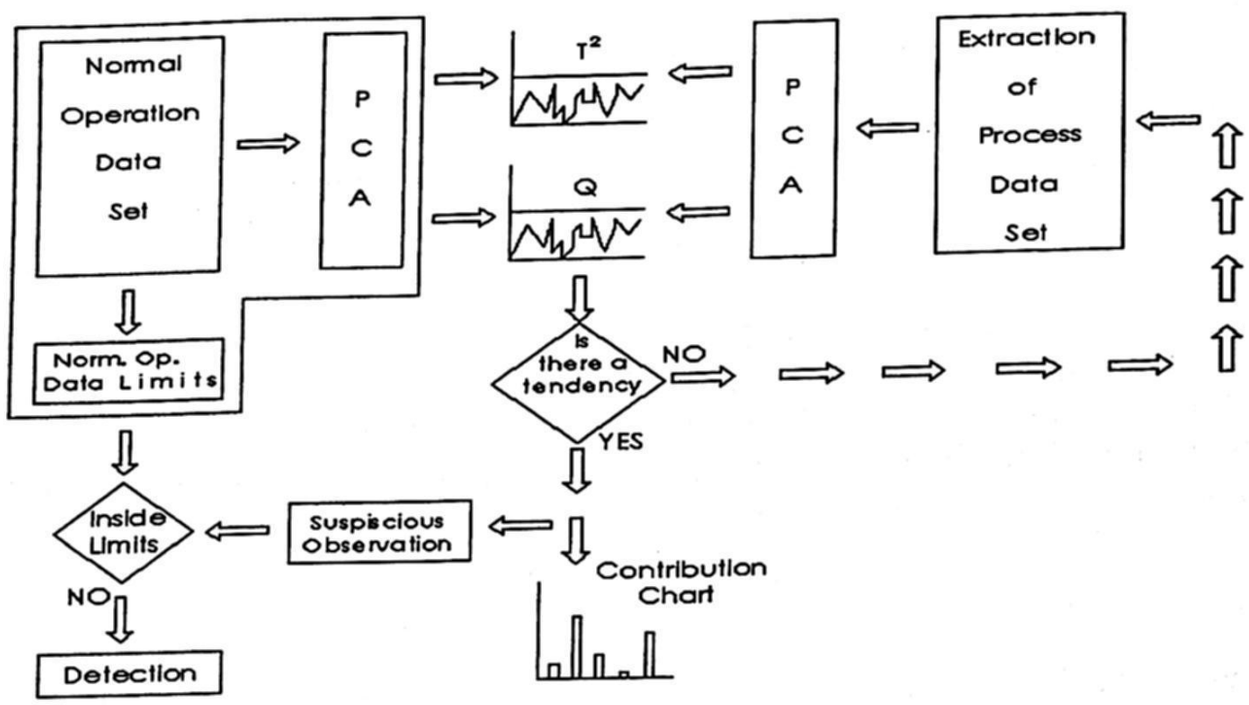


Fig. 2. General fault detection framework

The first step is to obtain a normal operation data set from system or process and a standardization procedure is carried out. From this data set a PCA model is obtained to see the relationships and to find out the correlations between variables, that at a first sight would be very difficult to notice. Then, a decrement in original data matrix dimension is achieved, allowing to work with less but enough data to describe the maximum variability of original system. The PCA model gives the loads and scores of the principal components corresponding to normal operation. With this loads and scores of normal operation, the value of the explained variance for each component as well as the boundaries

for Hotelling's T^2 and Q statistics are obtained (equations 4 and 6 respectively). They are the corresponding limits in control charts. In addition, the normal operation data limits are obtained following the steps mentioned in section 2.1. An adequation to the method described in [11] is being done. Instead of taking several signal transformation functions (STF) and so many single output functions (SOF), here it is taken just one STF and one SOF. Raw data is taken as the STF and the mean as the SOF, making it possible to obtain the minimum and maximum statistic limits that variables being monitored should have in normal operation.

For detection process, a process data set is extracted and analized as follows. A PCA model is built from the extracted process data set, and the scores of the principal components given by this model are plotted in control charts comparing them with the T^2 and Q statistics corresponding to normal operation model. If the chart does not exhibits a trend, another process data set is extracted to be analized. But if a normal operation T^2 or Q limit is violated or a trend is present, two actions are taken. 1)A contribution chart is done to find out which variable or variables has the major contribution to system's variability, and 2)the suspiscious observation is compared to the normal operation data limits previously obtained to verify if it is out of bounds.

4 Case Study

This section shows the performance of our framework in a simulation example. The simulation consists in the operation of an electric system formed by five subsystems. Each subsystem is simulated with different RL series circuits (see figure 3). A change within $\pm 10\%$ of the original values in each subsystem's components is considered as normal operation. An electrical current sensor is available such that each subsystem's current could be measured. After PCA a reduction to 2 variables was obtained.

The methodology proposed is applied as follows:

1. From normal operation history process data (electrical current in each subsystem), build PCA model and obtain T^2 and Q statistics as well as the minimum and maximum limits for each subsystem's current.
2. Take a test data set.
3. Build a PCA to the test data set and obtain a reduction on original dimensions.
4. Build and observe control charts for T^2 and Q statistics. If control chart detects a trend in a specific time instant go to 5), else go back to 2).
5. Build contribution chart and obtain the electrical current value for suspiscious time instant (sample).
6. Compare the suspiscious electrical current value with its normal operation limits obtained in 1) and detect which subsystem is in faulty mode.

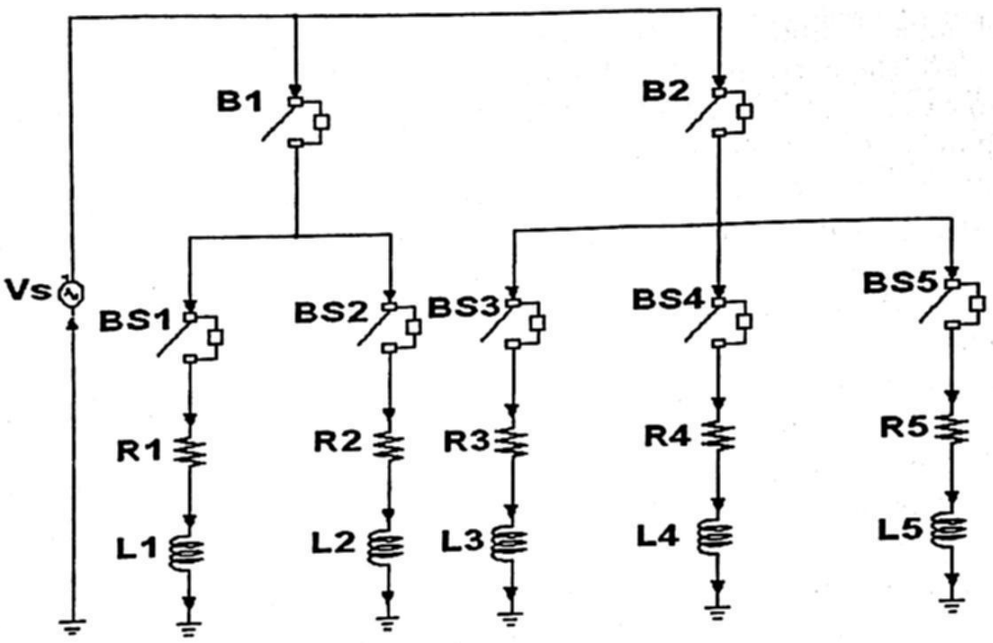


Fig. 3. Simulation of a system formed by five subsystems.

4.1 Single Fault

Several runs have been done to detect simple faults and obtain the effectiveness percentage in this task. A ramp input, a step input and a combination of both were simulated in different subsystems to see the performance of the methodology proposed. For instance, a simulation of a single fault present in subsystem 2 in which current decrements in steps of 0.9% is included in sample 80. Figure 4 shows how control charts depict a trend to pass its control limits.

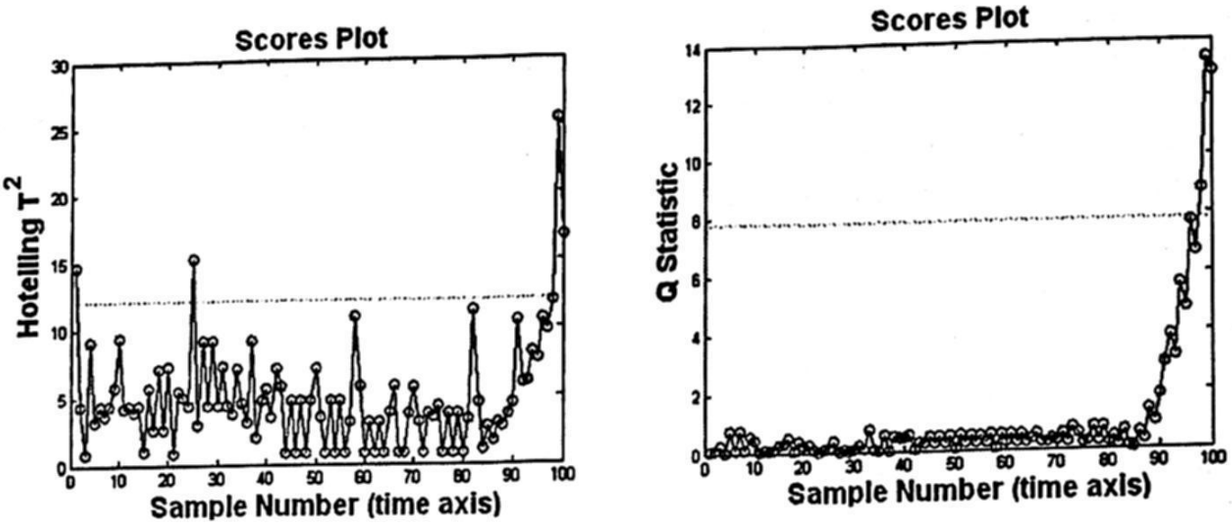


Fig. 4. Control charts for a decrement of 0.9% in subsystem's 2 current.

Figure 5 shows that variable 2 (subsystem 2) is the one that has the major variability of the system, indicating thus that subsystem 2 is probably in faulty mode. Then, when checking this suspicious observation against its corresponding normal operation limits, it is found that subsystem's 2 electrical current value has decreased under the lower current normal operation limit, having in this way detected the fault.

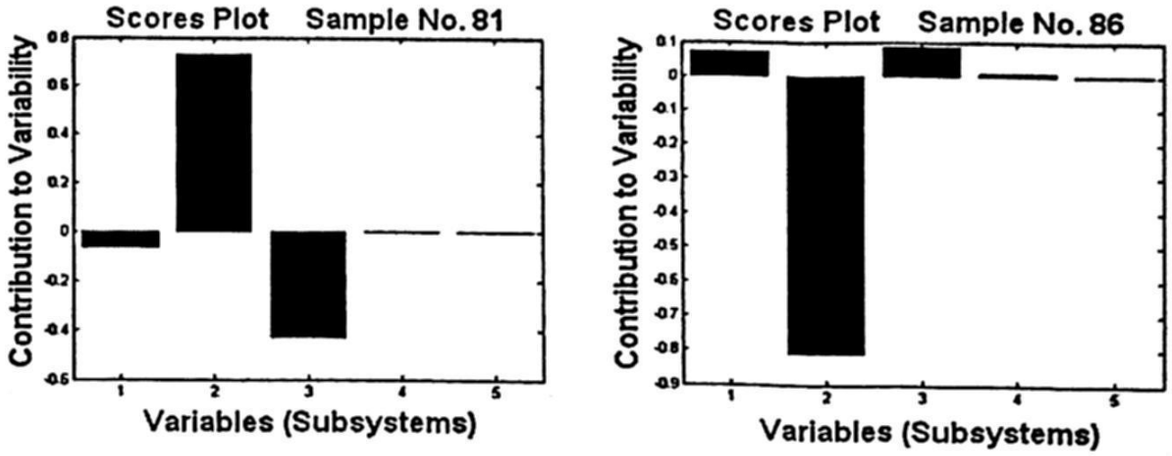


Fig. 5. Contribution charts for samples 81 and 86, indicating a probably problem in subsystem 2.

4.2 Multiple Fault

In the case of multiple fault, several runs have been carried out simulating a fault in two subsystems simultaneously. A ramp input in both subsystems, a ramp in one and a step in the other, a step in both, and a combination of a negative and positive ramp and step in one, the other or both. In this case contribution chart not always shows the real variables that possibly have problems. That is the reason explaining why the use of minimum and maximum limits for normal operation data plays an important role. As an example, a multiple fault is included in sample 80. Subsystem 3 having increments of 10% and subsystem 5 decrements of 0.9% in theirs corresponding electrical current normal operation value. Note that figure 6 depicts how the contribution chart does not show the real variables in multiple faulty mode, but it shows too how the implementation of statistic limits really does. Adittionally, the general performance of the proposal is shown in table 1.

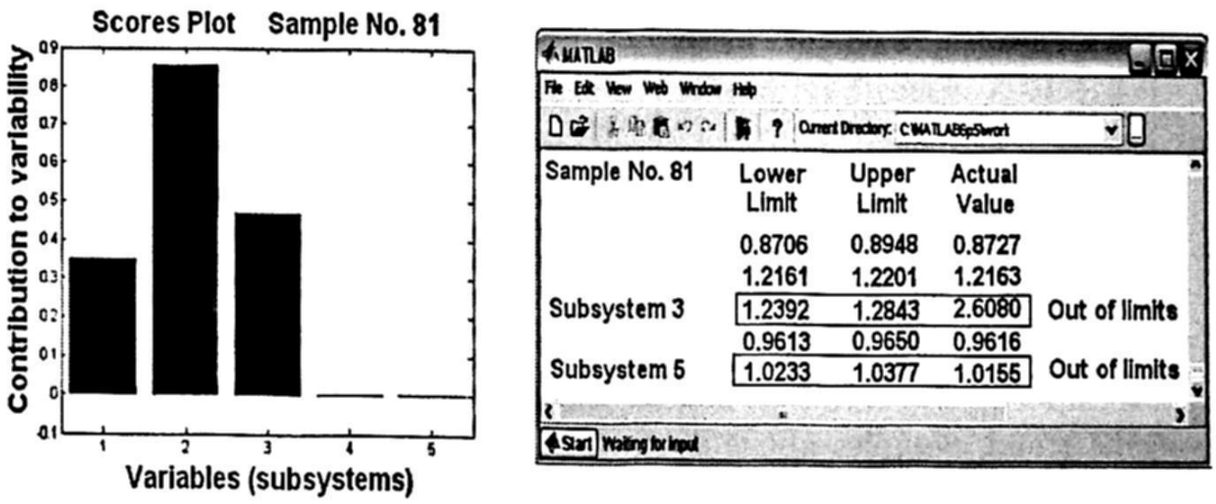


Fig. 6. Contribution charts for sample 81 (left) and the detection using the statiscal limits for the same sample (right).

Table 1. General performance of the proposal.

Fault	Number of Simulations	Percentage Detected
Single	50	98%
Multiple	50	96%

4.3 Measurement Noise

Table 2 shows the results for simulations taking into account different measurement noise magnitudes present in one, two or three subsystems. Figure 7 depicts the behaviour of control charts when it exists measurement noise of 0.1 magnitude present in subsystem 1.

Table 2. Performance of detection when measurement noise is present in one, two and three subsystems.

Noise Magnitude	One Subs.	Two Subs.	Three Subs.
0.0001	100%	100%	100%
0.001	100%	100%	85%
0.01	95%	85%	80%
0.1	90%	85%	75%

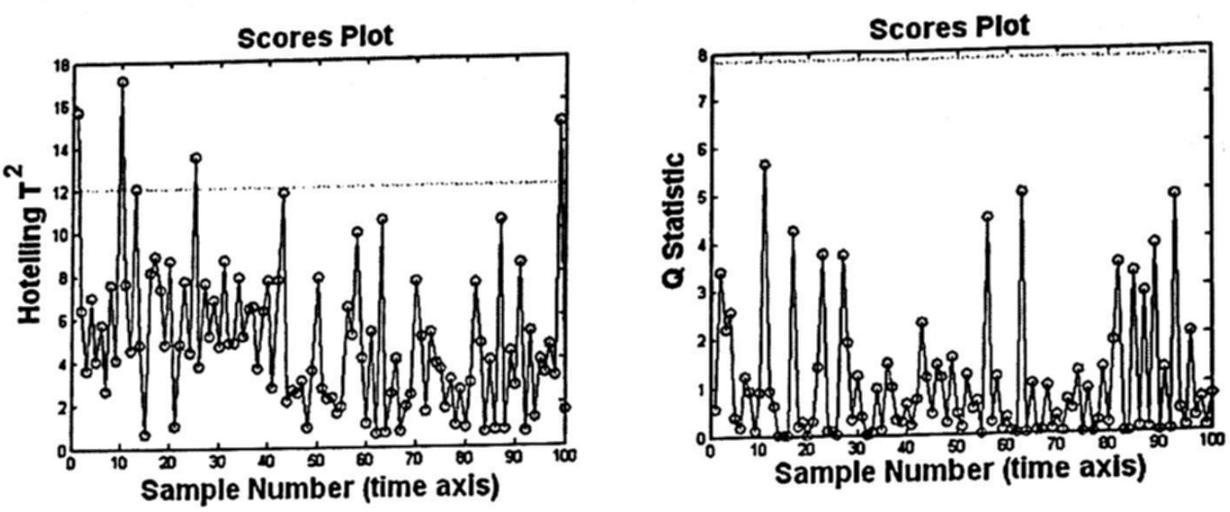


Fig. 7. Presence of measurement noise of 0.1 magnitude in subsystem 1.

It is important to note that none of the control charts shown in figure 7 has a specific trend. Samples above T^2 control chart limit are outliers. Note that all samples in Q control chart are below its limit, which has sense because in

this case noise does not brake the original correlation between variables. As T^2 is a measure of the multivariate distance between samples with respect to the center of data, it could detect a fault that keeps the correlation structure which could not be detected by Q. Q detects faults that violate mass or energy balance pointing out a correlation breakdown.

4.4 Comparison against two similar frameworks

To observe the general performance of our proposal, a comparison against two similar Process History Based fault detection methods has been carried out. We have chossen [10] and [11] as the comparison frameworks because of their use of PCA and Multi-dimension features extraction of signal based on machine learning respectively. Table 3 shows this general performance comparison.

Table 3. Comparison of the general performance of our proposal against two similar frameworks.

Detection of	PCA method	Machine Learning	Our Framework
Single Fault	√	√	√
Multiple Fault	NO	√	√
Measurement Noise	NO	NO	√
Process Noise	NO	NO	NO
Lack of Information	NO	NO	NO

From table 3 it could be observed that PCA used as itself offers a poor data analysis. It is observed that machine learning based method (ML) as well as ours framework offer multiple fault detection. Nevertheless the use of ML needs to be implemented for each measured signal which generates a big quantity of data to be analysed. Meanwhile our framework avoids this data explosion by mean of the use of PCA, control charts and the obtention of the normal limits operation just for the statistical mean of variable been measured. An additional advantage over the other two frameworks is that ours detect measurement noise.

5 Conclusions

This paper has presented a fault detection framework based on history process data. An advantage over model based methods is that this framework only needs a good historical data set of normal system operation, which in practice it is relativelly easy to obtain for computer controlled industrial processes. This proposal is easy to implement and to adapt because when original process changes, it is only required to modify the original data base instead of develop a new mathematical model from it. Another advantage is that the use of PCA model allows to work with less quantity of data, but keeping the original correlation between variables. It is important to note that the use of T^2 and Q control charts allows

to distinguish between the presence of a fault and the presence of measurement noise. Also this framework could be used as an early way for fault detection as shown in subsection 4.1 when a deviation of 0.9% on variable been measured was detected. Finally the use of minimum and maximum limits for comparisons between a suspicious sample and its normal values gives the detection of a single or multiple faults existing in the system.

References

1. V. Venkatasubramanian, R. Rengaswamy, K. Yin, S. Kavuri (2003): A review of process fault detection and diagnosis Part I, Part II and Part III. Computers and Chemical Engineering 27(2003) 293-311.
2. N. Zanzouri, and M. Tagina (2002): A Comparative Study of Hybrid System Monitoring Based on Bond Graph and Petri Net Modeling. 2002 IEEE SMC.
3. D. Du, X. Lou, and C. Wu (2005): Dynamic Model of FCCU and its Application in a Hybrid Fault Diagnosis System. 2005 International Conference on Control and Automation (ICCA 2005), June 27-29, 2005, Budapest, Hungary
4. W. Wang, X. Bai, W. Zhao, J. Ding and Z. Fang (2005): Hybrid Power System Model and the Method for Fault Diagnosis. 2005 IEEE/PES Transmission and Distribution Conference & Exhibition: Asia and Pacific Dalian, China
5. Q. Peter He, J. Wang, and S. Joe Qin (2004): A New Fault Diagnosis Method Using Fault Directions in Fisher Discriminant Analysis. TWMCC Texas-Winsconsin Modeling and Control Consortium. Technical report number TWMCC-2004-05. Department of Chemical Engineering. The University of Texas at Austin, Austin USA.
6. A. Barigozzi, L. Magni, and R. Scattolini (2004) : A Probabilistic Approach to Fault Diagnosis of Industrial Systems IEEE Transactions on Control System Technology, Vol. 12, No. 6, November 2004
7. W. Liang Cao, B. Wang, L. Ma, J.Zhang and J. Gao (2005) : Fault Diagnosis Approach Based on the Integration of Qualitative Model and Quantitative Knowledge of Signed Directed Graph. Proceeding on the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou, 18-21 August 2005
8. S. Gentil, J. Montmain and C. Combastel (2004) : Combining FDI and AI Approaches Within Causal Model Based Diagnosis IEEE Transactions on Systems, Man and Cybernetics. Part B: Cybernetics, Vol. 34, No. 5, October 2004
9. G. D. Gonzlez, R. Paut, A. Cipriano, D. R. Miranda, G. E. Ceballos (2006) : Fault Detection and Isolation Using Concatenated Wavelet Transform Variances and Discriminant Analysis. IEEE Transactions on signal processing, Vol. 54, No. 5, May 2006.
10. J. Liang, N. Wang (2003) : Fault Detection and Isolation Based on PCA: An Industrial Reheating Furnance Case Study. International Conference on Systems, Man and Cybernetics, 2003.
11. W. Shi, H. Yan, K. Ma (2005) : A New Method of Early Fault Diagnosis Based on Machine Learning. Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangahou, 18-21 August 2005.
12. R. Johnson, D. Wichern : Applied Multivariate Statistical Analysis. Prentice Hall, fifth edition.

Computer Networks and Web Services

Videoconference Transport in the IP Core Network of a 3G Mobile System using DiffServ

Ernesto Quiroz, Jesús Leal and Rogelio Rodríguez

Instituto Politécnico Nacional- Centro de Investigación y Desarrollo de Tecnología Digital
Av. del Parque No. 1310, Mesa de Otay, 22510 Tijuana, Baja California, México.
Phone:(664)623-1344, Fax:(664)623-1388

{eequiroz, jleal, rrodriguez}@citedi.mx
http://www.citedi.mx

Abstract. Video content is bound to occupy the greater percentage of the bandwidth among its multimedia kin (voice, web, etc.), and also to consume most of the processing resources of the IP-based networks carrying it [1]. Among video streaming applications (compressed video, web seminars, video clips, etc.), videoconference exhibits real time constraints, which sets tight time delivery and packet loss bounds to the network. These constraints are stressed in the mobile environment, where mobile terminal handoff between cells and break-up prone radio links put additional burden to the Quality of Service (QoS) mechanisms of the mobile network. In this work, a computer model of a UMTS release 5 IP core network is developed for the analysis of time delay, packet loss and traffic handling capacity of videoconference sessions. Results obtained indicate that the use of Diffserv for QoS provision can aptly handle a wide spectrum of situations.

Keywords: Streaming video, QoS, UMTS, Diffserv.

1 Introduction

New Third Generation (3G) mobile networks allow up to 2 Mbps for a non-moving nomadic terminal. At these data rates it is outright straightforward to transport multimedia services (video, multimedia messages, etc.). Nevertheless, Quality of Service (QoS) for real time applications imposes stringent bounds of delivery delay on the packet network. Video content is in growing demand in fixed networks, which boast ample to wideband channels and medium to high-speed processors; however, a mobile environment is limited by radio spectrum and mobile terminal processing power. 3G mobile systems (UMTS: Universal Mobile Telecommunications Systems, CDMA2000) show added shortcomings, stemming from the radio-link breakage possibility when a cell-to-cell handover or a steep decrease in signal strength occurs. Under these circumstances, it is of capital interest to characterize the videoconference carrying capacity of 3G systems. The main consideration is that if videoconference Quality of Service requirements are met, those of streaming video in general can also

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 281-289

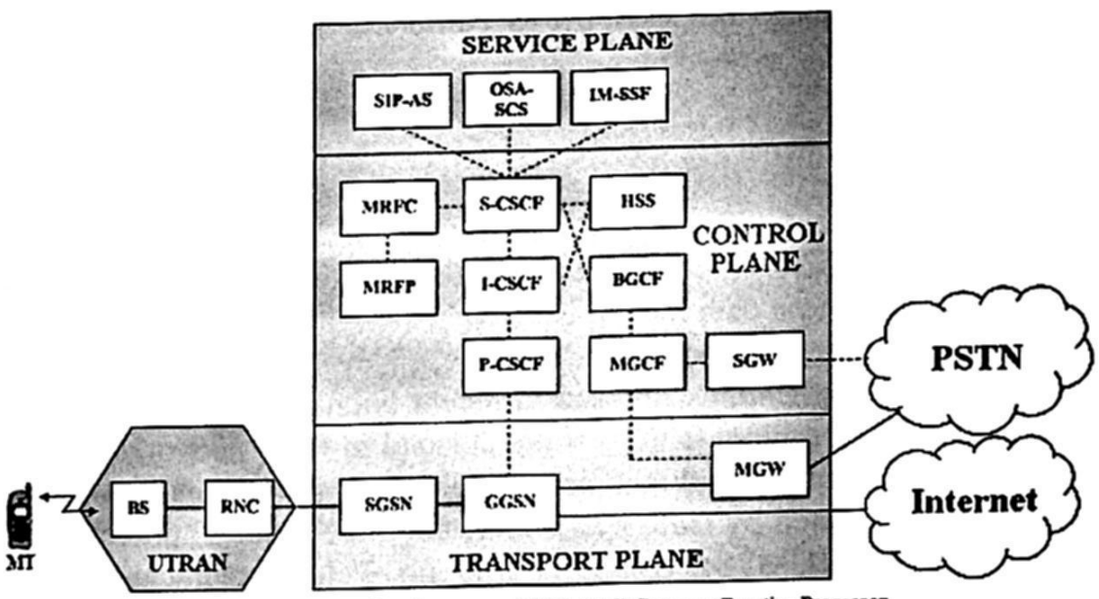
Received 12/02/07
Accepted 08/04/07
Final version 18/04/07

be satisfied. In this context, a UMTS core network model to analyze videoconference stream transport is presented. An H.263 video coder computer model is developed to obtain an accurate behavior of the videoconference data traffic. Time and packet loss constraints are taken care of by a DiffServ discipline in the UMTS IP core network. The effects of resource competition between videoconference and other multimedia streams are also considered.

The organization of the article is as follows. Section 2 explains the main functionalities of the UMTS, especially from the point of view of diverse media transport capacity and QoS provision. In section 3 an UMTS IP Core Network computer simulator is presented, as well as its configuration and operational details. Section 4 deals with the probabilistic representation of the H.263 videoconference traffic behavior. Section 5 puts forth a selected group of simulation outcomes, which help to characterize the limitation and capabilities of the UMTS IP Core Network. Finally, some conclusions are drawn up.

2 3G Mobile System Architecture

Figure 1 illustrates the UMTS release 5 architecture, as defined by the 3GPP (Third Generation Partnership Project). Doted lines represent signaling links, while solid lines indicate bearer channels. The service plain administers and executes the IP based multimedia services. The control plain performs the signaling functions necessary for session management. The transport plain provides the network resources to carry and route the users' data payload.



BGCP: Breakout Gateway Control Function
BS: Base Station
CGSN: Gateway GPRS Support Node
GPRS: General Packet Radio Service
HSS: Home Subscriber Server
IM-SSF: IM - Service Switching Function
I - CSCF: Interrogating - CSCF
MGW: Media Gateway
MGCF: Media Gateway Control Function
MRFC: Media Resource Function Controller
MRFP: Media Resource Function Processor
MT: Mobile Terminal
OSA - SCS: Open Service Access - Service Compability Server
P - CSCF: Proxy - CSCF
RNC: Radio Network Controller
S - CSCF: Serving - CSCF
SGSN: Serving GPRS Support Node
SIP - AS: Session Initiation Protocol - Application Server
SGW: Signaling Gateway

Fig. 1. UMTS Release 5.

There are two main functionalities [2]:

(a) UTRAN (UMTS Terrestrial Radio Access Network): Comprised of the base stations (BS or Node B) and the Radio Network Controller (RNC). Handles all the broadcasting functions to connect the Mobile Terminal (MT) and the Base Station, and takes care of mobility management.
(b) IP Core Network (IPCN): Provides and manages the channels to connect the UTRAN to the Internet and PSTN. It also provides end-to-end QoS guarantees.

Within the IPCN, various packet management mechanisms come into play. In accordance with the IP processing of real time applications, the Real Time Protocol (RTP) services videoconference organizing packets in UDP (User Datagram Protocol) [2] datagrams. The GPRS Tunneling Protocol (GTP) [3], [4], [5] communicates the RNC and the GGSN, setting up tunnels to conceal protocol handling of external networks. DiffServ [6], [7], supports QoS requirements by means of assigning priority levels to each data flow entering the edge router, thus specifying which flow will be processed first, second, and so on, at every router in the session path. Eight bit tag identifiers are inserted in every packet to advice the router processor of the packet priority.

3 UMTS IP Core Network Computer Model

Since we are interested on the effects the IPCN is going to exert over the videoconference traffic, it is the bearer conduits we need to focus on, which leads us to the simplified representation of the UMTS system shown in Fig. 2.

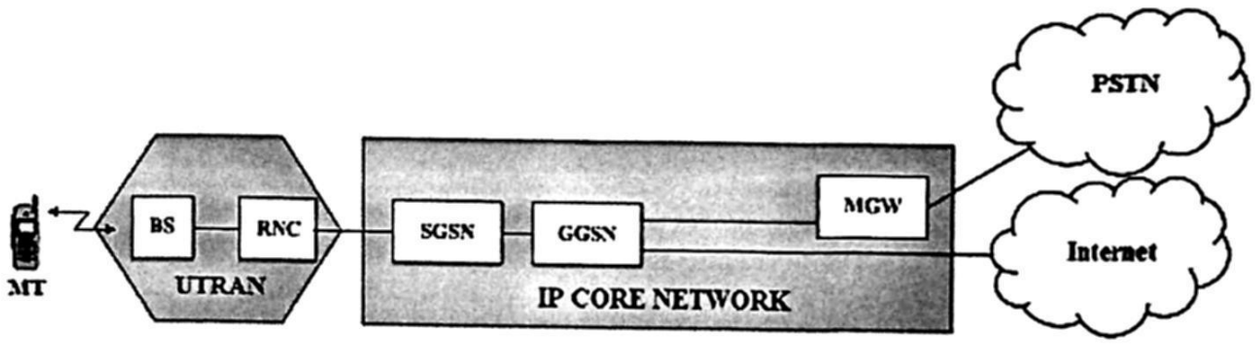


Fig. 2. UMTS simplified core network.

Using this guideline a computer model operating under the Internet Protocol was developed using the OPNET platform [8]. Figure 3 depicts the simulation model wherein each part of the simplified representation of figure 2 can be identified. As an extension to the Internet cloud, a second mobile network is introduced, which contains the complementary videoconference MT. Three additional MTs help shape the multimedia traffic. Two voice sources (MT_Voice) are used; one communicates with the PSTN, and the other with the mobile network B MT. The voice model represents an AMR (Adaptive MultiRate) codec [9] at 12.2 Kbps rate packetized in 20 ms

frames (32.5 bytes/frame). Call duration follows an exponential probability density function (t_{av} = 3 min.), with voice activity detection (talk 40%, silence 60 %).

In order to determine the combined effect the network and multimedia traffic will have over the videoconference stream, the MT_Load is configured to yield high volume packetized data streams into the IPCN at various data rates (42 Kbps to 8.4 Mbps) [10]. The MT_Video generates traffic at 66 Kbps average. Its representation and statistical behaviour is developed in the next section.

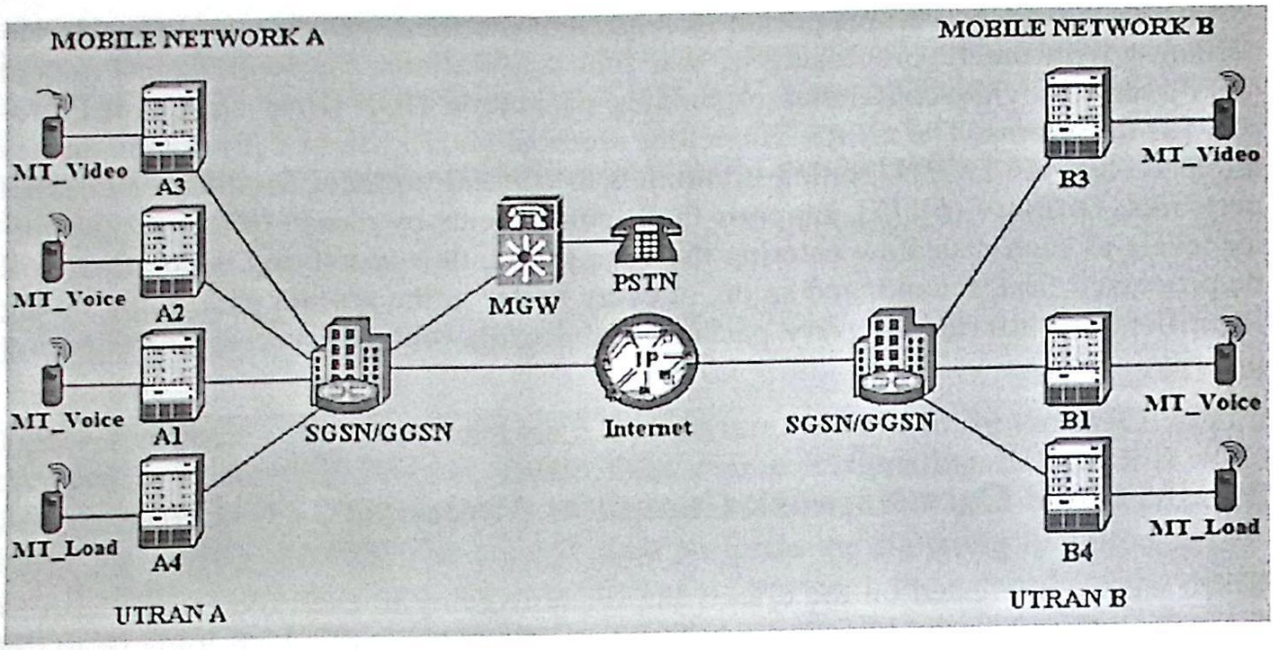


Fig. 3. UMTS-IPCN simulation model.

GTP tunneling is performed in the communications between the RNC and the SGSN/GGSN. The RTP protocol segments the videoconference traffic and introduces sequencing and timing information in each packet [7]. QoS is supplied by the Diff-Serv protocol, according to the levels assigned in Table 1.

Table 1. DiffServ classes assignment

Traffic Source	DiffServ Class	UMTS Class
UE_Video	AF41 (Gold)	Conversational (Video)
UE_Voice	EF (Premier)	Conversational (Voice)
UE_Load	BE (Best Effort)	Background

4 Videoconference Coder Algorithm and Application

An accurate representation of videoconference data flows was a central requisite for this work, to be able to characterize how the network elements, mechanisms, and other media affect its flow. Several probabilistic behaviour models of coded video data have been developed [11][12][13]. We adopted the Hanzo, et al. [14] model of

the H.263 ITU-T (International Telecommunications Union-Telecommunications) [15], [16], recommendation for videoconference. In particular the QCIF (Quarter Common Intermediate Format) format was selected, because its coding range spans from 32 to 384 Kbps, ideal for the bandwidth of 3G radio-channels.

The model is based on a 20 state Markov chain, where each state represents a range of bits/frame of coded video. The probability that k bits/frame will be generated in a specific state is governed by a Poisson probability density function, expressed by

$$\mathrm{P}(k,\lambda_i)=\frac{(\lambda_i T)^k \exp(-\lambda_i T)}{k!} \quad . \tag{1}$$

λ_i is the average number of bits generated in each state, T indicates the time duration of each cycle. Furthermore, λ_i is calculated as follows:

$$\lambda i = R\min + i\frac{R\max - R\min}{N} = R\min + i\Delta R \quad . \tag{2}$$

N is the number of states of the Markov chain; R_{max} and R_{min} correspond to the upper and lower limits of the bits/frame respectively. There is also a transition probability from state i to j $\{i = 1, 2...20\}$. Additional details can be found in [12].

In order to obtain a videoconference stream with an average rate of 66 Kbps the following values were adopted: $T = 10$ μs, $N = 20$ $(0 \leq i \leq N)$, $R_{max} = 3{,}400$ y $R_{min} = 1{,}000$ bits/frame, $D = 5{,}000$, $O = 3$, $\Delta R = 120$. The algorithm was implemented in Matlab, and run to obtain 3,000 frames, equivalent to 100 secs. of video play. Figure 4 graphics a short window of the results (frames 100 to 200).

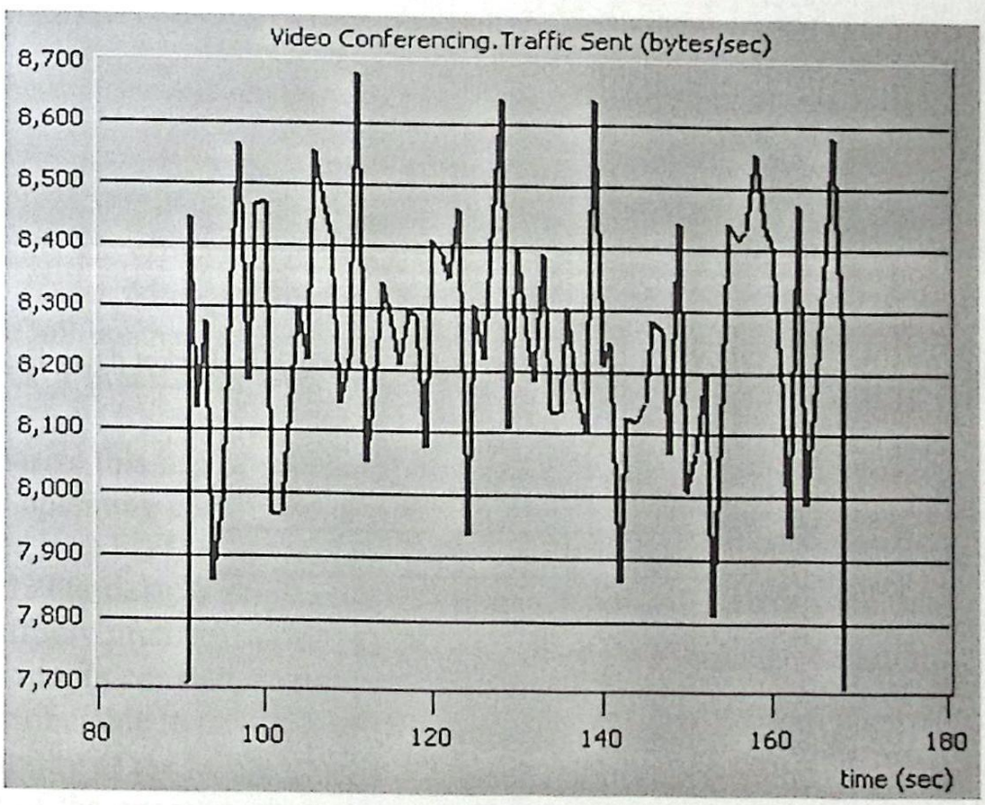


Fig. 4. H.263 coder data rate variation.

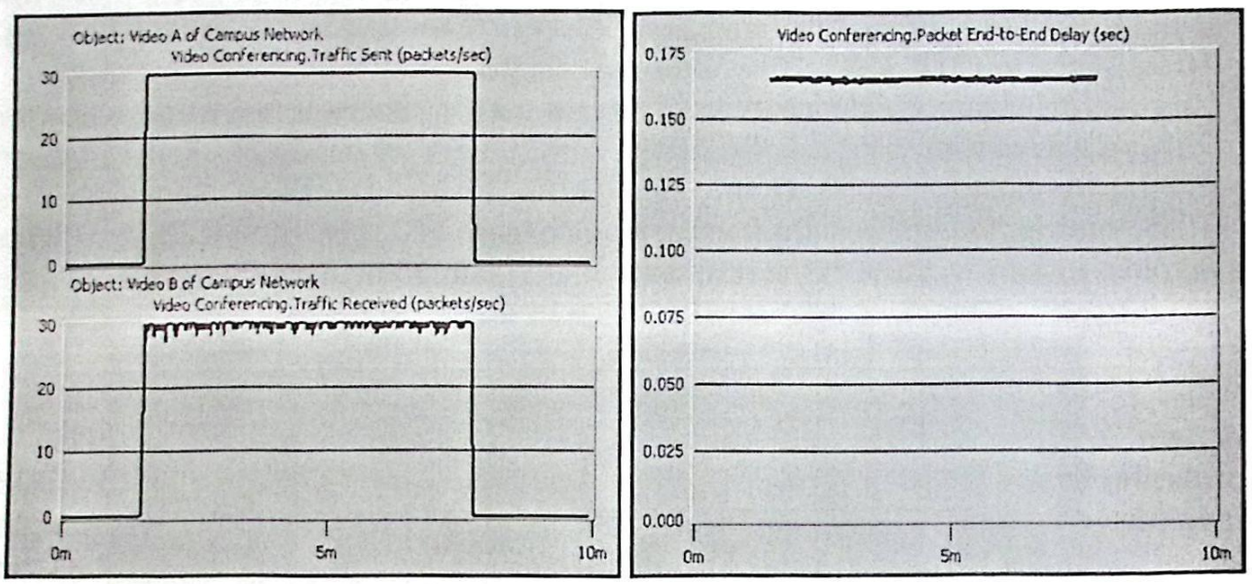
5 Performance Metrics

Over 30 different IPCN operating situations were configured and performed. Table 2 specifies two of them, whose output offers the more relevant information.

Table 2. Core Network settings

	MT_Load data rate	Internet link capacity	SGSN/GGSN processing capability
1	3.5928 Mbps	4.096 Mbps	10.0 Mbps
2	4.4298 Mbps	4.096 Mbps	12.5 Mbps

Figures 5a and 5b illustrate packet loss and delay obtained for operational setting 1, while Table 3 states exact figures for the same setting. Figures 6a, 6b, and Table 4 display corresponding information for setting 2.



a) b)

Fig 5. a) Videoconference stream packet loss in setting 1, b) Packet delay in setting 1

Table 3. Performance metrics in setting 1

Aspect	Value
Session duration	376 s
Packet sent	11277
Packet received	11160
Loss percentage	1%
Maximum delay	165.52 ms
Average delay	164.74 ms

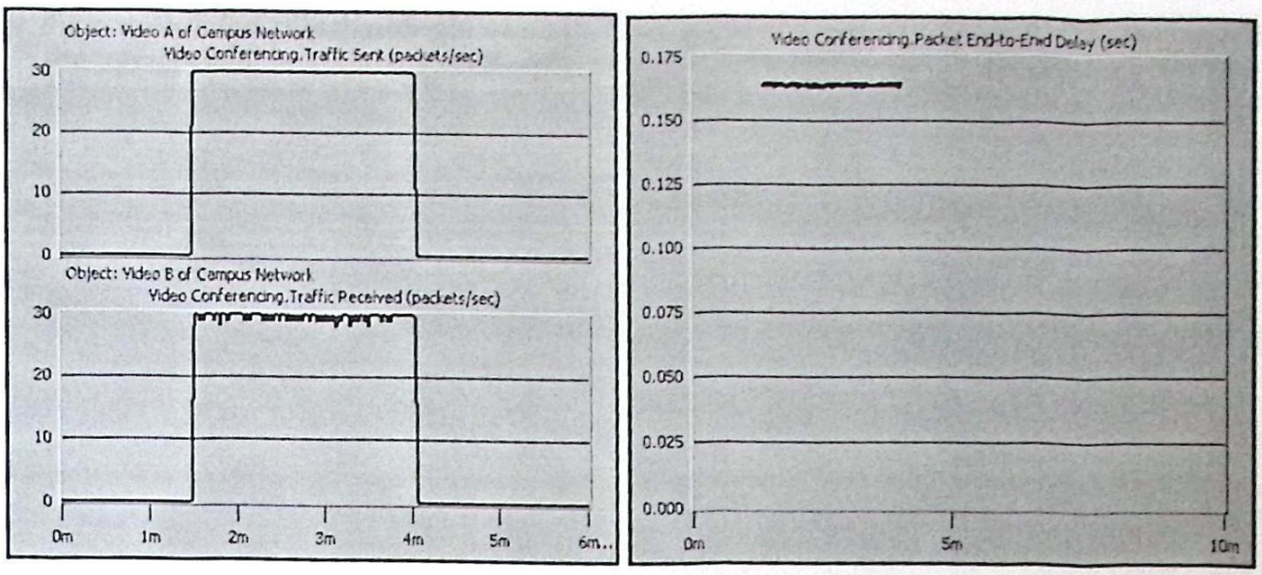


Fig 6. a) Videoconference stream packet loss in setting 2, b) Packet delay in setting 2

Table 4. Performance metrics in setting 2

Aspect	Value
Session duration	153 s
Packet sent	4592
Packet received	4545
Loss percentage	1%
Maximum delay	164.53 ms
Average delay	163.80 ms

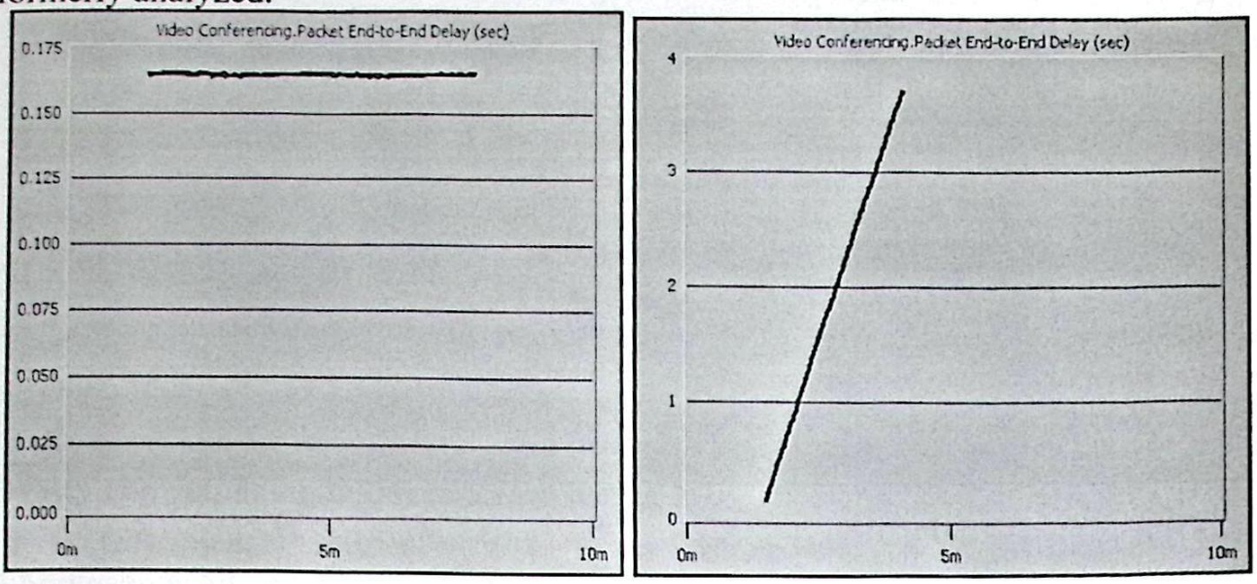
The outcomes from both settings are very similar in packet loss and time delay, fulfilling the generally accepted constraints: Packet loss ≤ 1%, Delay < 200 msec. This highlights the efficiency of the Diffserv protocol, because in spite that the load in setting 2 causes a congestion condition in the Internet link (4.4298 Mbps > 4.096 Mbps), the QoS mechanism, because of the priority assigned to the videoconference stream, gives it precedence over other traffic, yielding very good performance figures.

To emphasize this line of reasoning, two graphics are shown below, corresponding to the same operating conditions of Table 2, except for the Diffserv discipline which is substituted by a FIFO (First In First Out), in other words, without QoS support.

In case 1, the delay's range is within boundaries, practically the same obtained with the DiffServ protocol, since the Internet link and SSGN/GGSN router capacities are not overrun. In contrast, a considerable delay in the order of seconds is incurred in setting 2, attributable to the congestion condition mentioned above. The greater processing capability of the SGSN/GGSN router does not help improve the performance.

The rest of the operational configurations tested include a group of more relaxed situations, and other group where the aggregated traffic is roughly double than that of

the Internet link or the router capacity, overall showing consistent behaviour with the formerly analyzed.



a) b)

Fig 7. a) FIFO packet delay in setting 1, b) FIFO packet delay in setting 2

6 Conclusions

Packetized videoconference stands a challenge to any IP network carrying it, because of its low tolerance to delay and packet loss. Thus, it's of capital importance to characterize its behavior, especially in mobile networks.

A computer model of the UMTS Release 5 IP Core Network is developed to obtain key performance parameters of its treatment of coded videoconference, namely, time delay and packet loss. An accurate probabilistic model of the H.263 videoconference stream is also software developed.

The results obtained indicate that the IPCN with Diffserv is capable of handling the videoconference service within operational margins, even in the presence of heavy congestion in the network.

Acknowledgments. This work was financially supported by SIP-IPN grant 2007-0907 and COFAA-IPN Becas por Exclusividad.

References

1. Seeling P., Reisslein M. and Kulapala B., Network Performance Evaluation Using Frame Size and Quality Traces of Single-Layer and Two-Layer Video: A Tutorial, vol 6, IEEE Communications Surveys & Tutorials (2004) 58-78.

2. Camarillo G. and García-Martín M. A., The 3G IP Multimedia Subsystem (IMS): merging the Internet and the cellular worlds, John Wiley & Sons (2004).
3. Wisely D., Eardley P. and Burness L., IP for 3G: Networking Technologies for Mobile Communications, John Wiley & Sons (2002).
4. Chen J. C. and Zhang T., IP-Based Next-Generation Wireless Networks: Systems, Architectures and Protocols, John Wiley & Sons (2004).
5. Kaaranen H., Ahtiainen A., Laitinen L., Naghian S. and Niemi V., UMTS Networks: Architecture, Mobility and Services, John Wiley & Sons (2001).
6. http://www.ietf.org/html.charters/OLD/diffserv-charter.html
7. David Soldani, Man Li and Renaud Cuny, QoS and QoE Management in UMTS Cellular Systems, John Wiley & Sons (2006) 16, 70.
8. http://www.opnet.com
9. 3GPP TS 26.141 v6.0.0 (2004-12), Media formats and codecs. http://www.umtsworld.com/ technology/mediaformats.htm
10. Leal N. J., Modelo de núcleo de red IP de una red 3G para transporte de multimedia en tiempo real, Master's thesis, CITEDI-IPN, México (2006) 66-68.
11. Cosmas J., Petit G., Lehenert R., Blondia C., Kontovassilis K., Casals O., and Theimer T., A Review of voice, data and video traffic models for ATM, European Transactions on Telecommunications, vol. 5, (1994) 139-154.
12. Rose O. and Frater M., A comparison of models for VBR traffic sources in B-ISDN, in Proceedings of the IFIP TC6 Second International Conference on Broadband Communications, (Paris, France), , Chapman and Hall Ltd, London, (1994) 275-287.
13. D Heymann. and Lakshman T., Source models for VBR broadcast-video traffic, IEEE / ACM Transactions on networking, vol. 4 (1996) 40-48.
14. Hanzo L., Cherriman P. J. and Streit J., Wireless Video Communications: Second to Third Generation Systems and Beyond, IEEE Series on Digital & Mobile Communication,. (2001) 205-246.
15. ITU-T, Recommendation H.263: Video Coding for Low Bitrate communication, March 1996.
16. Minoru Etoh, Next Generation Mobile Systems 3G and Beyond, John Wiley & Sons (2005) 247.

Workflow Partitioning to Support Decentralised P2P Based Enactment

Tariq N. Ellahi, Francesco Nerieri and T. Kechadi

School of Computer Science and Informatics
University College Dublin
Dublin - Ireland
{tariq.ellahi, tahar.kechadi}@ucd.ie, nero@google.com

Abstract. Grid workflow management systems has been a focus of research in recent years and as a result a lot of systems have been developed. Most of the existing systems employ a centralised workflow enactment engine to coordinate the execution of workflow activities. This approach results in a single point of failure and increased execution overhead due to the existence of a single enactment engine. In this paper, we present a workflow partitioning mechanism which can enable decentralised enactment of the workflow activities by a number of distributed workflow engines coordinating in a peer-to-peer fashion. Initial Experimental results show the efficiency of the partitioning procedure.

1 Introduction

Grid computing [1] enables the aggregated utilization of heterogeneous and distributed resources such as computing elements, data sources, instruments and application services. With the help of grid systems, the scientific community is developing and deploying complex applications to manage and process large data sets, and execute scientific experiments on distributed grid resources. A workflow application is a collection of computational activities to be executed according to the user-specified invocation order to accomplish some specific goal. Workflow application is represented as a directed graph. Activities are modeled as vertices of the graph and dependencies among activities are represented as edges. Workflow management systems define, create and manage the execution of workflow applications through the use of software, running on one or more workflow engines, which is able to interpret the process definition, interact with workflow participants and, where required, invoke the use of IT tools and applications. Workflow management systems have been used by the business community to manage business processed for a very long time. Recently, workflow support in the grid systems was adopted and a number of grid workflow management systems have been developed during the last few years.

The existing grid based workflow management systems employ a centralised workflow enactment mechanisms. In this scenario, a single workflow execution engine coordinates the execution of all the activities in the workflow graph. This architecture not only introduces a single point of failure but it severely restricts

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 291-304

Received 13/02/07
Accepted 08/04/07
Final version 23/04/07

the scalability of the system especially in the grid landscape where thousands of users will be using the system. One solution to overcome this problem is to distribute the workflow enactment tasks among a number of workflow engines. A couple of grid based workflow management systems adopt the decentralised enactment approach. In order to be able to support the decentralised enactment of workflow applications, workflow graph must be partitioned and each partition is assigned to a different workflow engine. This approach will improve the scalability of the system and result in increased efficiency and reduced execution time.
In this paper we present a workflow partitioning mechanism that will enable the grid system to execute the workflow application without relying on a centralized point of command and control. This approach is used in the grid workflow management system which is part of the Data Grid Environment & Tools (DGET) [2][3][4] grid middleware. The rest of the paper is structured as follows: Related work on workflow partitioning is discussed in section 2. Sections 3 and 4 present general information about workflow terminology and the lifecycle of a workflow application. Detailed explanation of the workflow partitioning procedure with examples are given in section 5. Analysis and evaluation of the workflow partitioning procedure is explained in section 6. Section 7 concludes the paper and presents future research direction.

2 Related Work

There are several workflow management systems that has be developed during last few years [5][6][7][8][9][10][11][12]. Almost all of the systems referenced except[5][6] employ centralised workflow enactment and thus no partitioning procedures are applied. There are a couple of workflow management systems that have been developed with the same objectives as our systems i-e decentralised enactment of the workflow application. Workflow partitioning was applied in the Pegasus [5] system but the objective and the method used were different than our partitioning procedure. The objective of the workflow partitioning system in Pegasus was efficient mapping of the workflow activities onto grid resources. The Workflow application is partitioned by levels of the graph thus each level was isolated as a separate partition. This approach didn't take into account the data dependencies among the activities which is a crucial factor in order to enable the system to support decentralized workflow enactment.

The second workflow management system that incorporates workflow partitioning is Askalon [13]. In Askalon partitioning is performed as an optimization of the workflow execution. Contrary to Pegasus and our system, the partitioning procedure is applied after the scheduling of the workflow activities has been performed. Its partitioning procedure is quite straight forward in Askalon. All the activities that have been scheduled for execution on the same grid site are put into one partition. This approach while useful to some extent still might not result in efficient enactment of the workflow application. Workflow partitioning before the scheduling process can take into account the data dependencies

among workflow activities. Thus different scheduling strategies can be applied to different partitions based on the nature of the activities contained in the partition.

3 Workflow Lifecycle

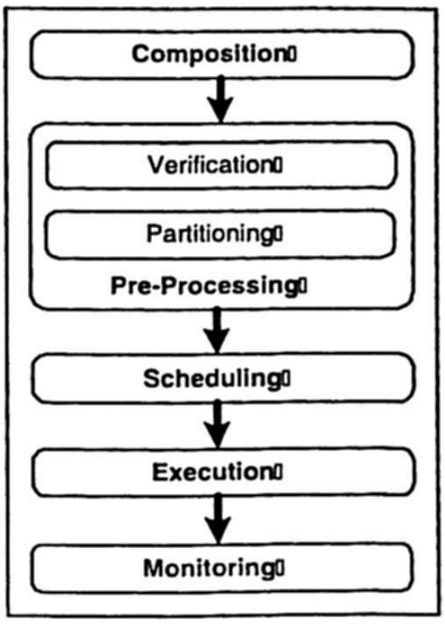


Fig. 1. Workflow Lifecycle

In this section we describe the typical lifecycle of a workflow application. Every workflow application passes through a number of phases during its lifetime. There are five such phases in the workflow application lifecycle. Following subsections explain each of the phase in details.

3.1 Composition

Workflow lifecycle starts with the composition of the workflow application using the workflow specification language or other high level visual composition tools. User specifies all the activities at an abstract level without specifying the grid related details. Beside specifying the activities, user also defines the dependency relations among the activities. After the workflow specification, the workflow is parsed and an internal java representation of the workflow graph is generated by the workflow engine. This java representation is used during the later phases of the workflow lifecycle. The workflow parsing process can also be taken as a pre-verification procedure where workflow is checked for errors missing values in the workflow specification.

3.2 Pre-Processing

After the composition phase, workflow graph is pre-processed. The pre-processing phase is composed of two sub-phases: Verification & Partitioning. These sub-phases are explained in the following paragraphs.

Verification As described in the previous section, workflow specification is parsed and checked for syntactical anomalies in the composition phase. In the verification phase, workflow graph is checked and verified for any structural conflicts like deadlocks, livelocks and lack of synchronization etc. The verification procedure is explained in a separate section later in this paper.

Partitioning After the successful verification of the workflow graph, the partitioning phase is entered. Since the focus of our work is peer-to-peer based enactment of the workflow, workflow specification must be decomposed into a set of distinct activities that can be enacted by a number of cooperating workflow engines. This decomposition of done in the partitioning phase. Section * describes the partitioning procedure at great length.

3.3 Scheduling

Workflow application specified during the composition phase did not include any grid related information e.g resource which are going to be used to execute the activities. During the scheduling phase, the activities are mapped onto grid resources taking into account the QoS parameters specified in the activity definition. The scheduling is performed at the partition level. Services of middleware components like Resource Discovery are used to find matching resources that can be used for activity execution.

3.4 Execution & Monitoring

After successful scheduling of workflow activities, the execution of the workflow activities commence. Along with the execution of the activities, monitoring of the execution is also performed. if anything goes wrong with the execution of some activity, the monitoring component triggers the fault management component so the execution rescue can be performed.

4 Workflow Partitioning

Workflow partitioning is the process of decomposing the workflow graph into a number of smaller non-empty, non-overlapping set of activities. Different partitions can be assigned to different workflow engines so that workflow can be executed in a distributed and decentralized fashion. Partitioning procedure takes into account the minimization of the inter-partition dependencies. This results in a number of distinct advantages:

- Distributed and decentralized workflow enactment in a Peer-to-Peer fashion. This also eliminates a centralized workflow engine overlooking and coordinating the execution of overall workflow application.
- Reduced Scheduling and job submission overhead
- Reduced Communication Overhead
- Increased fault tolerance and resilience
- Improved execution efficiency

4.1 Notations

The following notations will be used in this document to explain different aspects of workflow management.

- $\mathcal{W}$: Workflow graph with vertices representing the activities and edges representing the data dependencies between the activities
- $\mathcal{A}$: Set of activities in the workflow $\mathcal{W}$ (Vertices of the graph)
- $\mathcal{D}$: Data Dependencies between activities in the workflow $\mathcal{W}$ (Edges of the graph)
- $\mathcal{P} = \{p_1 \dots p_n\}$: Set of partitions of the workflow $\mathcal{W}$
- $a_i \longrightarrow_d a_j$: Data dependency between activity a_i and a_j where $a_i, a_j \in \mathcal{A}$
- *pred*(a): Direct predecessor(s)/parents of an activity a $\in \mathcal{A}$
- *succ*(a): Direct successor(s)/children of an activity a $\in \mathcal{A}$
- s: Start activity of the workflow $\mathcal{W}$
- s_p: Start activity of a partition p $\in \mathcal{P}$
- *level*(a): Level of the activity a in the workflow $\mathcal{W}$
- *ldiff*(a_i, a_j) = *level*(a_j) - *level*(a_i): Level Difference between activities (a_i) and (a_j)

4.2 Partition Properties

All the partitions in the workflow exhibit the following properties:

- Non-Empty: $\forall p_i \in \mathcal{P}, \mid p_i \mid > 0 \wedge \sum_{i=0}^{n} \mid p_i \mid = \mid \mathcal{A} \mid$
- Non-Overlapping: $\forall p_i, p_j \in \mathcal{P}, p_i \cap p_j = \emptyset \wedge \bigcup_{i=1}^{n} p_i = \emptyset \wedge \bigcup_{i=1}^{n} p_i = \mathcal{A}$
- No Intermediate Data Dependencies:

$$\forall a_i, a_j \in \mathcal{A}, \nexists\ a_i \longrightarrow_d a_j \text{ where } a_j \in p_i \in \mathcal{P} \wedge a_i \notin pi \wedge a_j \neq s_{pi}$$

4.3 Partitioning Algorithm

The general motivation behind the workflow partitioning is to divide the workflow activities into a number of disjoint sets of activities. By doing so, each subset of activities can be assigned to a separate workflow engine for enactment and therefore eliminating the existence of a centralized workflow engine overseeing the execution of overall workflow application. One critical factor in the partitioning is to make sure that such isolated activities in a partition must have

minimum possible dependencies and interaction with the rest of the activities in other partitions. Activities in a workflow applications are organized into different patterns, therefore, the functionality of the partitioning algorithm is to search for a number activities arranged in some patterns with least possible synchronization points. Before we describe the partitioning algorithm, it will be necessary to explain the different patterns, our algorithm will be looking for in the workflow specification. Partitioning algorithm will try to identify the following types of patterns:

Activity Pipeline: Activities arranged in a sequential spatial order. This pattern is similar to an execution pipeline where activities are started one after another without interacting with the rest of the activities in the workflow. Partitioning algorithm looks for a set of activities which are arranged in a sequential order with no synchronization points during the execution of the pipeline. However, pipeline can begin with a synchronization point which mean, the workflow engine will have to wait till the synchronization is achieved and after that, it can continue executing the rest of the activities in the pipeline without reliance on other activities in the workflow. Figure 2 shows an example of a pipeline, example on the left side show a pipeline without any synchronization point and the one on the right with a synchronization point at the beginning of the pipeline.

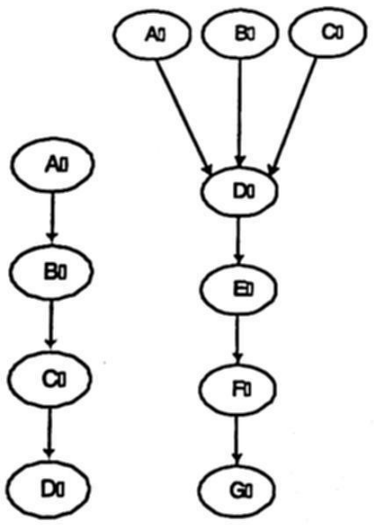


Fig. 2. Activity Pipeline

Parallel Block: The second type of activity pattern the partitioning algorithm tries to identify is a set of activities which can be executed independently of each other and from the rest of the activities in the workflow. Such activities exist when the execution flow reaches a split point and a number of activities are started that can be run in parallel. Different execution flows started by a split point will have to converge/synchronize at some stage later during the execution of the workflow. There can be different execution flow scenarios between a split

point and joint point. We identify the following scenarios and our algorithm treats each scenario in a different fashion.

Immediate Synchronization: This is the case where multiple instances of the same activity are started to achieve SPMD like scenario. All the instances synchronize at the next level. Another example could be where a number of different activities are started at the same time and they synchronize at the next level. Activities in this scenario can be isolated in a partition and executed by a separate workflow engine. Figure 3a is an example of this pattern.

Delayed Synchronization: In this scenario, the split point triggers a number of different execution flows that involves executing multiple activities in a pipeline or further split points can occur. These different execution flows converge at a later stage. The partitioning algorithm tries to find different patterns between the split and the join point and isolate them in different partitions. Figure 3b is an example of this pattern.

Hybrid: In this scenario, split point can instantiate a number of instances of the same activities and some execution flows like pipeline. The algorithm isolates the parallel activities in one partitions and the pipeline can result in a separate partition. Figure 3c is an example of this pattern.

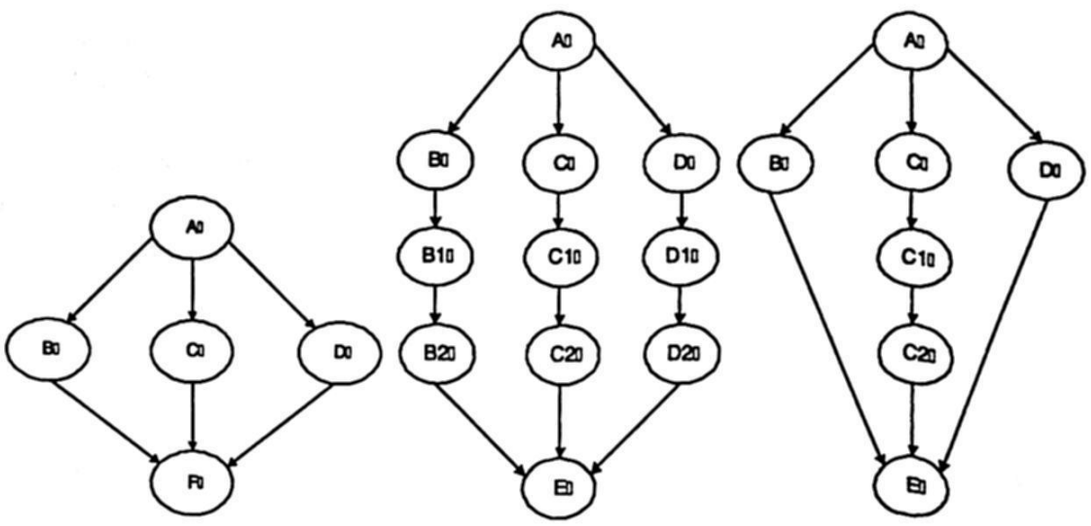


Fig. 3. Parallel Acivities

Iteration Block: Most of the workflow models are based on Directed Acyclic Graph (DAG) model. In such models loops are not allowed. There can be scenarios where some activity needs to be executed iteratively, therefore, our specification model allows loops in the workflow specification. Activities in an iteration block are isolated in a separate partition. Splitting activities in a loop into multiple partitions will result in dependencies among partitions, therefore, a separate partition is created for each loop definition specified in the workflow model. Iteration blocks are identified with the help of language constructs. This is done

before the partitioning algorithm is applied on the workflow specification. Figure 4 gives and example of iteration blocks both in graphical format and in XML format.

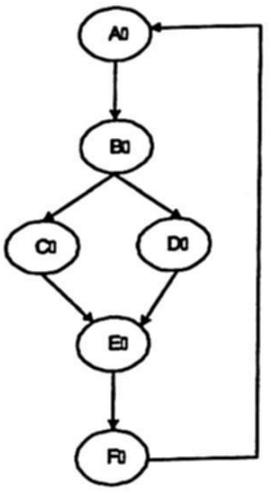


Fig. 4. Iteration Block

Conditional Execution: Another special case in the workflow specification is the conditional execution of the activities. In this execution splits into one of many possible execution flows depending on some conditional expression evaluated during runtime. As with the iteration blocks, conditional execution blocks are isolated into a partition of their own. Conditional execution blocks are identified with the language constructs from the workflow specification model before applying the partitioning algorithm.

Partitioning algorithm is given below:

```
PROCEDURE Partition (W)
QUEUE = {s}
while QUEUE ≠ ∅ do
  BLOCK = {}
  Get ν, First Node in the queue;
  if ν instanceOf Partition then
    remove ν from QUEUE;
    append succ(ν) to QUEUE;
  else if | succ(ν) | = 1 && ldiff((ν), succ(ν)) = 1 && | pred(succ(ν)) | = 1
  then
    remove ν from QUEUE;
    append ν to BLOCK;
    CALL extractPipeline(succ(ν));
  else if | succ(ν) | > 1 AND | pred(ν) | > 1 then
    remove ν from QUEUE;
    append succ(ν) to QUEUE;
```

```
    replace ν with a Partition
  else
    CALL extractParallel(ν);
  end if
end while
END Partition

PROCEDURE extractPipeline(startNode κ)
if κ instanceOf Partition then
  append succ(κ) to QUEUE;
else if | succ(κ) | = 1 AND | pred(κ) | = 1 then
  append κ to BLOCK;
  CALL extractPipeline(succ(κ));
else if | succ(κ) | > 1 AND | pred(κ) | = 1 then
  append κ to BLOCK;
  if succ(κ) NOT IN QUEUE then
    append succ(κ) to QUEUE;
  end if
  returnAFG(x + y)/2
else if | succ(κ) | = 1 AND | pred(κ) | > 1 then
  if κ NOT IN QUEUE then
    append κ to QUEUE;
  end if
  return
end if
replace nodes in BLOCK as a Partition in W
END ExtractPipeline

PROCEDURE extractParallel(startNode κ)
tempBlock = {};
append κ to tempBlock;
if succ(κ) NOT IN QUEUE then
  append succ(κ) to QUEUE;
end if
let δ = pred(κ);
while TRUE do
  ω = next node in the QUEUE;
  if pred(ω) ≠ δ then
    break;
  else if | succ(ω) | = 1 && ldiff((ω), succ(ω)) = 1 && | pred(succ(ω)) | = 1
  then
    remove ω from the QUEUE;
    CALL extractPipeline(ω);
  else
    remove ω from the QUEUE;
```

append ω to tempBlock;
append $succ(\omega)$ to the QUEUE;
end if
end while
replace nodes in the tempBlock as Partition in $\mathcal{W}$
return
`END extractParallel`

4.4 Partitioning Example:

Figure 5 and 6 gives two example workflow and their partitioned versions. Workflow graph structure in figure 5 is protein annotation workflow[14] and figure 6 is a neuro-science workflow[15]. The reason for taking these two as examples is because these workflow structures represent two different type of workflow graphs. Workflow in figure 5 is a regular workflow graph with regular data dependencies but the workflow graph in figure 6 is an irregular workflow with arbitrary data dependencies.

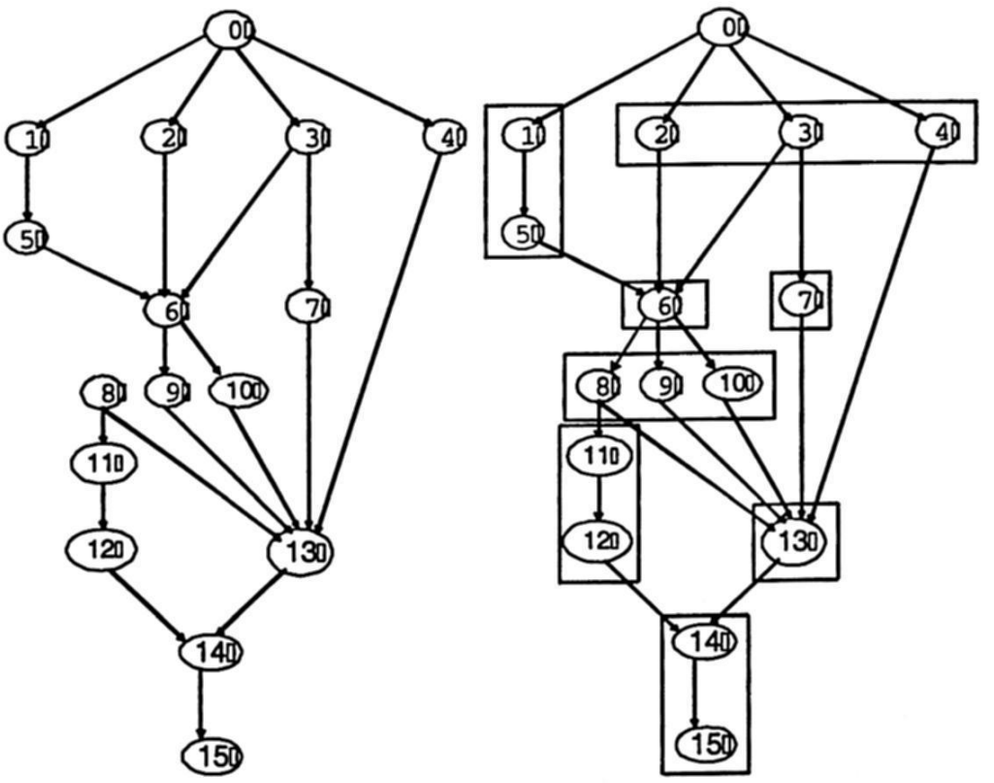


Fig. 5. e-protein Workflow Example

5 Analysis & Evaluation

This sections focuses on the analysis and evaluation of the workflow partitioning procedure. Initial experimental results are presented. Table1 shows the experimental setup and the machines used to run the experiments. Figure 7 and 8

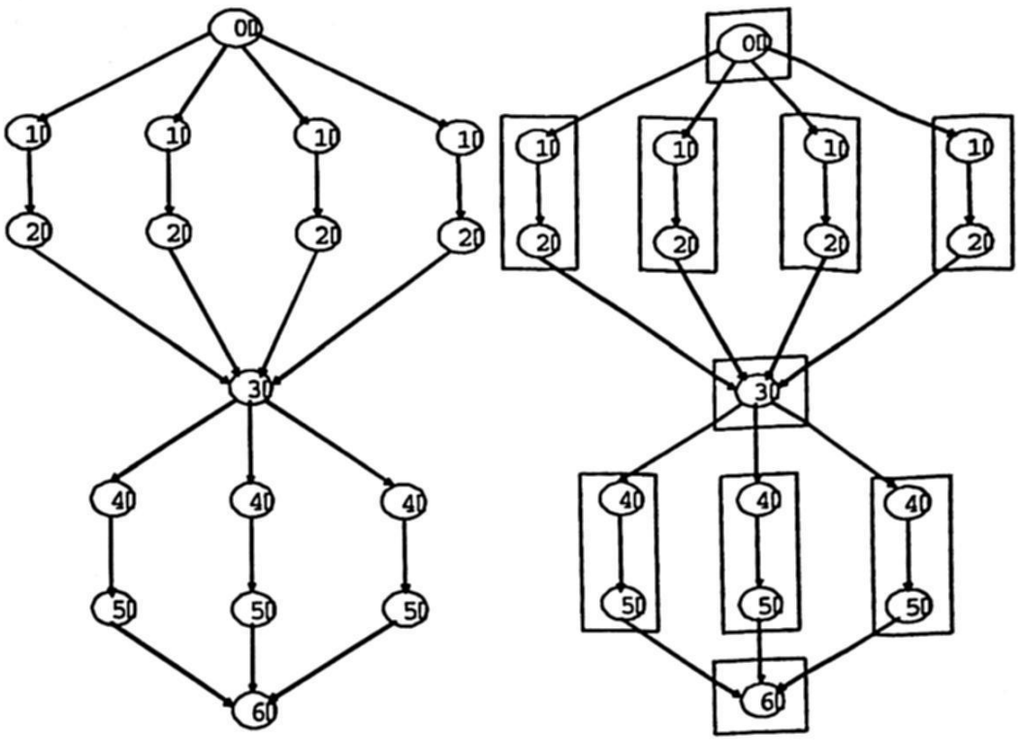


Fig. 6. fMRI Workflow Example

Table 1. Experimental Setup

Nr.	Machine	CPUs
1	bonnat.ucd.ie	4
2	pcrgcluster.ucd.ie	8
3	radostina.ucd.ie	2
4	bwhdzia.ucd.ie	1
5	tellahi.ucd.ie	1

shows the impact of partitioning process on the overall execution of the workflow application. Workflow partitioning doesn't add much overhead to the execution time of the workflow execution. As can be seen from figure 7, the partitioning phase constitutes 6% of the entire execution time of the workflow application but this cost of execution time spent in partitioning has its influence on the execution speed of the workflow application. Figure 8 depicts the execution times of a workflow application both in partitioned and non-partitioned scenarios. As can be seen the non-partitioned case takes more time to complete where the execution time is reduced to almost half when partitioning is performed. The overhead comes in the time spent to coordinate the execution of workflow activities by a single centralized entity. All the coordination messages has to be routed through the centralized enactment engine therefore increasing the execution time of overall workflow application.

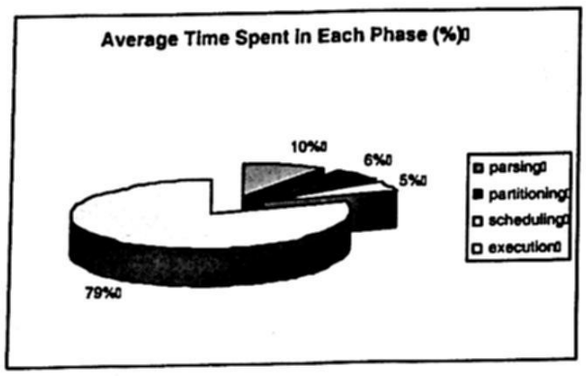


Fig. 7. Execution Time in Lifecycle Phases

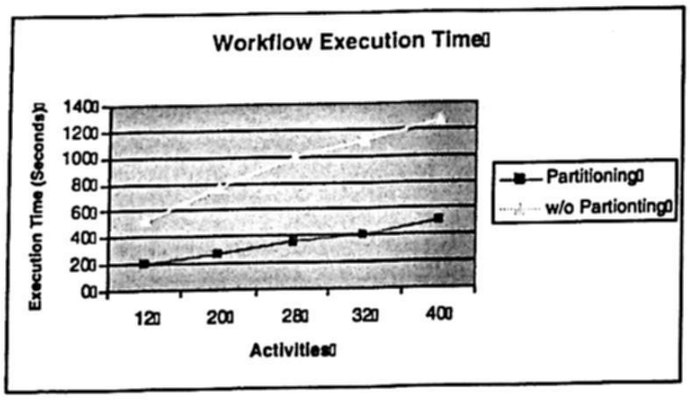


Fig. 8. Workflow Execution Time

Figures 8 and 9 depicts the partitioning behaviour when applied on different types of workflow graphs. Figure 8 represents the partitioning behaviou on regular and irregular workflow graphs. Since there are arbitrary data dependencies

in irregular workflow graphs, this results in a larger number of partitions being created especially some partitions where there is only one activity. This activity is typically an activity which has random data dependencies and can not be included in a serial or parallel block. In the sample workflow graphs, the number of activities were approximately the same but the structure of graphs were completely different. As a result, a regular graph is partitioned into half the number of partitions compared to the irregular workflow graph.

Figure 9 shows the partitioning behaviour on workflow graphs with majority of activities arranged in pipeline or parallel layouts. A workflow graph with majority of activities existing in pipeline form results in more number of partitions because activities in a pipeline can not have data dependencies on activities in other partitions. Whereas, in the second case, parallel activities do not have any impact and there are less number of partitions in the second case.

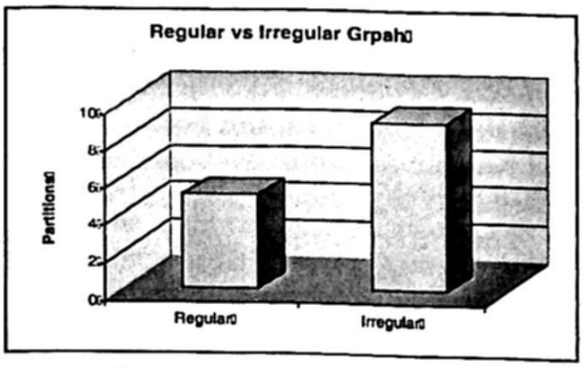


Fig. 9. Regular vs Irregular Workflow Partitioning

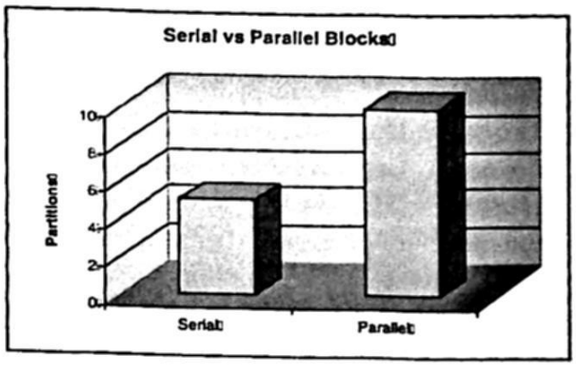


Fig. 10. Serial vs Parallel Block Partitioning

6 Conclusion & uture Work

In this paper we presented a workflow partitioning mechanism which can be used to split a workflow graph into a number of partitions. Each partition can be executed by a different workflow enactment engine. This approach eliminates a central entity coordinating the execution of entire workflow application. Experimental results show that partitioning a workflow application results in improved efficiency and reduced execution time for the workflow application. In the future, we would like to investigate the partitioning of different types of very large scale workflow involving thousands of activities. Another important research direction could be to ascertain whether partitioning is useful in all scenarios.

References

1. I. Foster and C. Kesselman, editors. *The Grid: Blueprint for a Future Computing Infrastructure.* Morgan Kaufmann Publishers, USA, 1999.
2. L. McDermott T. Kechadi T. N. Ellahi, B. Hudzia. A java based architecture of p2p-grid middleware. In *The 2006 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 2006)*, 2006.

3. L. McDermott T. Kechadi T. N. Ellahi, B. Hudzia. Entity management and security in p2p grid framework. In *7th International Conference on Computing (CORE 2006)*, 2006.
4. T.N. Ellahi B. Hudzia, L. McDermott and T. Kechadi. Entity based peer to peer in data grid environments. In *17th IMACS World Congress, Paris, France*, 2005.
5. Mei-Hui Su James Blythe Yolanda Gil Carl Kesselman Gaurang Mehta Karan Vahi G. Bruce Berriman John Good Anastasia Laity Joseph C. Jacob Daniel S. Katz Ewa Deelman, Gurmeet Singh. Pegasus: a framework for mapping complex scientific workflows onto distributed systems. *Scientific Programming Journal*, 13-3:219 – 237, 2005.
6. Juergen Hofer Farrukh Nadeem Francesco Nerieri Stefan Podlipnig Jun Qin Duan Rubing Mumtaz Siddiqui Hong-Linh Truong Alex Villazon Marek Wieczorek Thomas Fahringer, Radu Prodan. Askalon: A development and grid computing environment for scientific workflows. In *Workflows for eScience, XXX (Eds.), Springer*, 2006.
7. Jia Yu and Rajkumar Buyya. A novel architecture for realizing grid workflow using tuple spaces,. In *5th IEEE/ACM International Workshop on Grid Computing (GRID 2004, Nov. 8, 2004, Pittsburgh, USA), IEEE Computer Society Press*, 2004.
8. Francine Berman, Andrew Chien, Keith Cooper, Jack Dongarra, Ian Foster, Dennis Gannon, Lennart Johnsson, Ken Kennedy, Carl Kesselman, John Mellor-Crummey, Dan Reed, Linda Torczon, and Rich Wolski. The GrADS Project: Software support for high-level Grid application development. *The International Journal of High Performance Computing Applications*, 15(4):327–344, 2001.
9. Junwei Cao, S. A. Jarvis, S. Saini, and G. R. Nudd. Gridflow: workflow management for grid computing. In *3rd IEEE/ACM International Symposium on Cluster Computing and the Grid CCGrid 2003*, pages 198–205, 2003.
10. Chad Berkley Dan Higgin Bertram Ludäscher, Ilkay Altintas. Scientific workflow management and the kepler system. *Concurrency and Computation: Practice & Experience, Special Issue on Scientific Workflows*, 2005.
11. Matthew Addis M. Nedim Alpdemir Justin Ferris Kevin Glover Carole Goble Antoon Goderis Duncan Hull-Darren Marvin Peter Li Phillip Lord Matthew R. Pocock Martin Senger Robert Stevens Anil Wipat Tom Oinn, Mark Greenwood and Chris Wroe. Taverna: Lessons in creating a workflow environment for the life sciences. *Concurrency and Computation: Practice and Experience*, Volume 18 Issues 10:1067 – 1100, 2005.
12. David Churches, Gabor Gombas, Andrew Harrison, Jason Maassen, Craig Robinson, Matthew Shields, Ian Taylor, and Ian Wang. Programming Scientific and Distributed Workflow with Triana Services. *Concurrency and Computation: Practice and Experience (Special Issue: Workflow in Grid Systems)*, 18(10):1021–1037, 2006.
13. Thomas Fahringer Rubing Duan, Radu Prodan. Run-time optimization for grid workflow applications. In *7th IEEE/ACM International Conference on Grid Computing (Grid'06), (C) IEEE Computer Society Press*, September, 2006.
14. Darlington J O'Brien A, Newhouse S. Mapping of scientific workflow within the e-protein project to distributed resources. In *UK e-science all-hands meeting, AHM 2004, Nottingham, UK*, 2004.
15. Yong Zhao, Michael Wilde, Ian Foster, Jens Voeckler, Thomas Jordan, Elizabeth Quigg, and James Dobson. Grid middleware services for virtual data discovery, composition, and integration. In *MGC '04: Proceedings of the 2nd workshop on Middleware for grid computing*, pages 57–62, New York, NY, USA, 2004. ACM Press.

A Service-Oriented Architecture for Collaboratively Browsing the Web

Guillermo de Jesús Hoyos-Rivera[1], Giner Alor-Hernández[2], Roberta Lima Gomes[3], Roberto Willrich[4] and Jean-Pierre Courtiat[5]

[1]Department of Artificial Intelligence, School of Physics and Artificial Intelligence Universidad Veracruzana, Xalapa, Veracruz

[2]Division of Research and Postgraduate Studies, Instituto Tecnologico de Orizaba, Orizaba, Veracruz

[3]Informatics Department, Universidade Federal do Espírito Santo, Vitoria, Brasil

[4]Informatics and Statisticis Department, Universidade Federal de Santa Catarina, Florianópolis, Brasil

[5]Laboratory of Analysis and Architecture of Systems-CNRS, Toulouse, France

ghoyosr@gmail.com, galor@itorizaba.edu.mx, rgomes@inf.ufes.br, willrich@inf.ufsc.br, courtiat@laas.fr

Abstract. Service-Oriented Architectures (SOA) transform the ways in which the applications are created in a distributed environment work. Commonly, these applications are based on Web Services technologies. Web Services allow integration and collaboration through Internet standards. Recently, SOA has emerged as paradigm to develop collaborative systems, specially, Collaborative Web Browsing. Collaborative Web Browsing (co-browsing) aims at extending currently available Web browsing capabilities in order to allow several users to "browse together" on the Web. A co-browsing system should provide all the facilities required for allowing users to establish and release, in a very simple and flexible way, browsing synchronization relations as well as interactions with continuous media presentations embedded within Web pages. This paper presents the design, modeling, and implementation of the co-browsing system called CoLab from a point of view of SOA. CoLab provides all the functionalities required for allowing users to collaboratively browse the Web. CoLab presents a service-oriented architecture where the functionalities for co-browsing are performed.

Keywords: Collaborative Web browsing, continuous media, Service-Oriented Architecture, synchronization, Web Services.

1 Introduction

The World Wide Web (WWW) is a large distributed collection of documents connected by hypertext links. Web browsers are the basic tools for accessing and displaying these documents. Although this collection of documents can be concurrently accessed by several users, Web browsers are basically single-user tools.

Advances in Computer Science and Engineering
Research in Computing Science 27, 2007, pp. 305-317

Received 23/02/07
Accepted 08/04/07
Final version 21/04/07

Accordingly, users are isolated when browsing the Web since they have no way of sharing online their browsing activities with other users. A great effort must be made to allow a group of users to share their browsing activities (i.e. the pages they are visiting). Collaborative Web browsing overcomes this problem by allowing users to "browse together". In this paper, we consider a co-browsing system as a tool for allowing users to browse Web pages together in co-browsing sessions while establishing/releasing browsing synchronization relations among them as they wish.

This co-browsing system is based on a Service-Oriented Architecture (SOA) which describes a software architecture that defines the use of loosely coupled software services. Within our SOA, we provide a brokering service which uses Web Services technologies and can interoperate with other systems or software agents. We believe this way to proceed opens new possibilities in collaborative work since it breaks the currently existing isolation of users associated with Web browsing activities. As a result, collaboration relations can dynamically emerge as users browse the Web, discover new material, and share it online with other users, adding in this way a new dimension to the Web browsing paradigm.

However, there are several requirements that a co-browsing solution must meet. We believe that one of the most important ones is to provide flexible capabilities for organizing co-browsing sessions. Such an organization defines which users are authorized to follow a link and when and which user(s) should automatically retrieve a given resource. Most current co-browsing solutions adopt two types of organization for a co-browsing session: unmanaged or centralized. In an unmanaged organization, any member can follow a link while the other members will follow it automatically.

This way of working could turn the co-browsing session uncontrollable for groups of more than three users. Conversely, in a centralized organization, each session has a leader who controls the browsing actions. This organization type is only suitable for co-browsing sessions where the browsing actions of the leader must be followed by all the other session members. An alternative proposal for the organization of co-browsing sessions is allowing dynamic organization of session.

Here, session members can dynamically reorganize the co-browsing session in workgroups. A workgroup is composed by one or more session members whose browsing activities are synchronized. Workgroups can be dynamically created and modified. Therefore, beyond the centralized organization (where all the session members compose one workgroup), our solution allows creating a permanently or temporally decentralized organization. Workgroups can be temporally decentralized, and later, some of them can be merged together. This approach allows implementing the concept of "divide to conquer" [1], which is very important in Computer Science.
In this paper, we propose a co-browsing system called CoLab [2]. This co-browsing system is based on a simple and powerful synchronization model supporting a dynamic organization of a co-browsing session. The proposed model offers a simple mechanism allowing session members to create and release synchronization relations among them.

2 Service-Oriented Architecture of CoLab

SOA is a new approach to application development that requires people to work and think more cohesively and collaboratively than before. SOAs are based on the notion of services, which are high-level software components that include Web Services.

Web Services have attracted a lot of attention over the past years as a means of building and deploying software to simplify development and systems integration. Web Services are ideal for application integration and collaboration of internal systems or for linking software components over the Internet. Web Services technologies are based on open standards recommended by the World Wide Web Consortium (W3C). In this sense, we propose a co-browsing system following the SOA basic principles: 1) Integration, 2) Discovery and, 3) Publish. This co-browsing system is called CoLab. Some internals of CoLab are built on SOA. CoLab delivers a collaborative environment that includes capabilities such as co-browsing session, document and Web content management. In the service-oriented architecture of CoLab, there are two main components: 1) the CoLab "proxy server" and 2) the CoLab "client" which are described below. The general architecture is depicted in Fig. 1. The CoLab proxy server acts as a mediator between the website (where the requested Web pages are hosted) and the users of our system in order to manage co-browsing sessions.

This proxy server is composed of four main modules namely: 1) a "session manager"; 2) a "broker"; 3) a "browsing manager"; and 4) a "MediaSync manager." Additionally, it has an "integration manager".

The "session manager" manages the co-browsing session itself. This module offers the authentication and authorization functions based on the co-browsing session specification defining the default initial page, the available roles and their associated passwords, and the eventual existing privileges that can be associated with each of them. Roles are used as a way to allow some users to have privileges on other users when creating synchronization relation. The main component of the "session manager" is the "synchronization module," which treat all the synchronization actions and guaranteeing the overall consistency of the synchronization state.

Whenever a synchronization relation is created, the involved users' browsing activities get synchronized as well as the playing of continuous media (eventually embedded in the website).

The "broker" receives any browsing request from the user and asks the "session manager" to verify whether the request should be satisfied. This decision depends on certain conditions, such as the current synchronization state of the user or some other condition specified in an additional module integrated to CoLab (e.g., an access control module). The broker is proposed to be implemented as a Web Service-based brokering service. The general architecture of this module is depicted in Fig. 2. The broker is built by the following components:

1. **Service Registry** is the mechanism for registering and publishing information about services supported by CoLab. In this sense, we used a private UDDI [3] node which is an industry initiative to create a platform-independent, open framework for describing, discovering, and integrating Web Services.

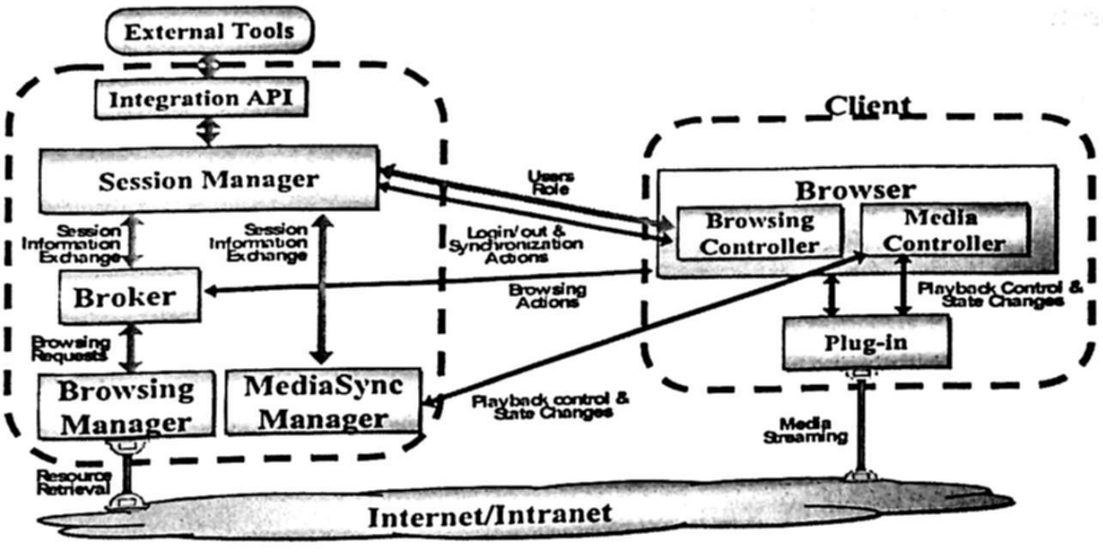


Fig. 1 Collaborative web browsing architecture.

2. **Discovery Service** is a component used to discover Web Services implementations. These Web Services can be obtained from the private UDDI node. Inside the discovery service, there is a query formulator which builds queries that will be sent to the registry service. This module retrieves a set of suitable services selected from the previous step and creates feasible/compatible sets of services ready for binding. The discovery service uses sophisticated techniques to dynamically discover Web Services and to formulate queries to UDDI nodes.
3. **Dynamic Binding Service** is a component that binds compatible Web Services. The binding of a Web Service refers to how deep is the degree of coupling with other Web Services. For instance, the technology of one Web Service provider might be incompatible with that of another even though the capabilities of both of them match with some requirements. In this sense, the module acts as an API wrapper that maps the interface source to a common interface supported by CoLab.
4. **Dynamic Invoker** transforms data from one format to another. This component can be seen as a data transfer object which contains the data (i.e. request or response) flowing between the requester to the provider applications of Web Services. We propose the use of Web Services Invocation Framework (WSIF) that is a simple Java API for invoking Web Services, no matter how or where the services are provided [4]. WSIF allows stubless or completely dynamic invocation of a Web service, based upon examination of the meta-data about the service at runtime. It also allows updated implementations of a binding to be plugged into WSIF at runtime, and it allows the calling service to defer choosing a binding until runtime.

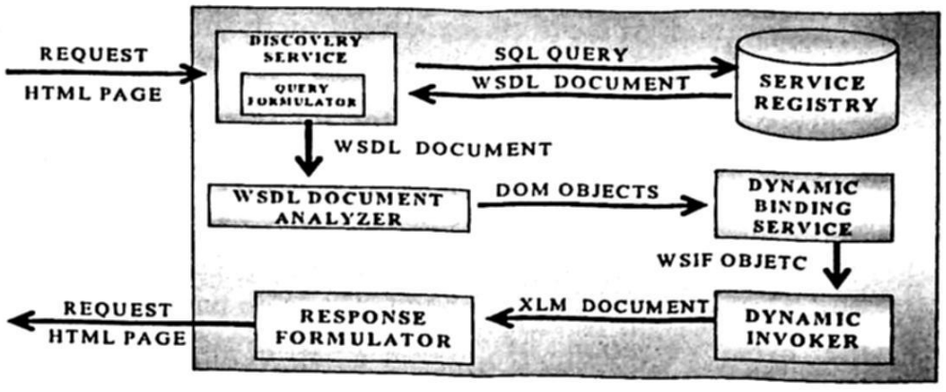


Fig. 2 Brokering service architecture.

5. **WSDL Document Analyzer** validates WSDL documents. In this context, this component reports the operations, input and output parameters, and their data types in a XML DOM tree. We propose the use of WSDL4J [5] to convert the XML DOM nodes in Java objects. It facilitates the creation, representation and manipulation of WSDL documents. WSDL4J API is an IBM reference implementation of the JSR-110 specification (Java API's for WSDL).
6. **Response Formulator** receives the responses from the suppliers about a requested product. This module retrieves useful information from the responses and builds a XML document with information coming from the service registry and the invocations' responses. This XML document is presented in HTML format using the Extensible Stylesheet Language (XSL).

The "browsing manager" carries out all the tasks related to the retrieval of the resources requested by the users. This includes three main components that interact in order to satisfy incoming browsing requests.

1) The "retrieval" module is responsible for retrieving every requested resource. They can be retrieved directly from the Web server specified in the requested URL or from the cache module. In the first case, the retrieval module sends the Web page to the translation module in order to modify it before sending the response to the user's browser, as well as to the cache module.
2) The "translator" module is responsible for modifying on the fly every retrieved Web page. This is necessary to allow our system to track the users' browsing actions. This translation is also required to include the necessary controls for synchronizing the continuous media presentations eventually embedded in these Web pages. The translation consists mainly of adding some control parameters specific to CoLab to each hyperlink definition in the retrieved Web page. When the Web page has embedded media presentations, this module modifies the HTML code in order to detect plug-in state changes and to notify this to CoLab.
3) The "cache" module corresponds to the implementation of a basic cache system, which is mainly used, but not only, for satisfying requests coming from synchronous users in order to improve the performance of the system. We assume that when a synchronous user browses, the requested resource has been previously retrieved by the asynchronous user, so it is faster to retrieve the

already translated version of the Web resource directly from the cache rather than retrieving it from the original server and retranslating it at each time.

The "session manager" is also responsible for interacting with the "integration manager" that is intended to provide an API allowing CoLab to be extended with new functionalities, such as an access control system, or to be integrated to other collaborative tools or integration environments, such as LEICA [6]. The "MediaSync manager" takes charge of all the tasks related to the presentation control of the eventual continuous media presentations embedded in Web pages. Its main function is to guarantee the synchronization of audio/video presentations (streamed or downloaded) by forcing the same presentation state in all synchronous users' plug-in.

As detailed in [7], this module maintains the current state of each continuous media presentation in the session based on "state change" messages sent by the "media controller" and controls the presentations states by sending "playback control" messages to the synchronous users.

The "browsing controller" and "media controller" are two modules present at the client side (see Fig 1). The "browsing controller" is the component in charge of establishing a connection with the CoLab proxy server. Through this connection, the users' browsers receive the commands to display Web pages whenever they are synchronized with another user. The "browsing controller" also provides users with all synchronization controls allowing creating and releasing synchronization relations.

The "media controller" controls and synchronizes the continuous media presentation in the current Web page for all users of a workgroup. This module does the following three functions:

1) records the state of each audio/video presentation;
2) captures state changes of the continuous media presentations, treats them locally, and then informs them to the "MediaSync manager";
3) receives playback control messages from the "MediaSync manager" and executes the playback control.

The "media controller" prevents synchronous users from executing any playback control action. In this case, the playback control is done by the "MediaSync manager" via "playback control" messages.

In next section, we present the operational behavior of CoLab through a typical case of study.

3 Operational Behavior of CoLab

In order to graphically illustrate the operational behavior of our proposal, we present in Fig. 3 the case of a typical browsing action performed by an asynchronous user and the resulting synchronization with another user.The first step consists of the request of a resource expressed by a user (1), which is treated directly by the "broker." Next, the "broker" contacts the "session manager" to ask it whether the user can retrieve the requested resource (2). If so, the "broker" sends the request to the "retriever" (3), which asks the "cache module" if that resource is already in the cache (4). Let us assume that this is not the case, so the resource is retrieved directly from the original

Web server (5–6), and if it is identified as an HTML resource, it is sent to the "translator" in order to be modified (7). Once the resource has been translated, it is sent back to the "retriever" (8) and also to the "cache module" for storing purposes (8–9). The "retriever" then sends the resource back to the "broker" (10), which sends it to the user who has made the request (11). Once the previous steps have been completed, the "broker" asks the "session manager" to synchronize this browsing action for all the users who are currently synchronized with the user who has just executed the browsing action (12). Then, the "session manager" sends a message to the browser of every synchronous user present in the same workgroup (13). Each browser will then separately make its own request for the indicated resource (14), which will be sent again to the "broker". The "broker" asks the "retriever" (15) for the retrieval of the resource, which itself asks the "cache module" to verify whether the resource is cached (16). Since the resource has already been stored in the cache, and this browsing action is the consequence of the synchronization of a browsing action, it is retrieved directly from the cache (17) and sent back to the "retriever" (18), which sends it back to the "broker" (19), finally satisfying the user's request (20). Fig. 3 also shows the behavior of the playback control of continuous media presentations. For instance, once a Web page containing a continuous media presentation is loaded, if the asynchronous user clicks the "play" button, the "media controller" sends the state change (21) to the "MediaSync manager." TheMediaSync manager module updates the presentation state and sends all users the "prepareToPlay" playback control (22 and 23).When this module receives a "state change" message from all users (24 and 25) indicating the "ready-ToPlay" state, it sends all users the "play" playback control (26 and 27).

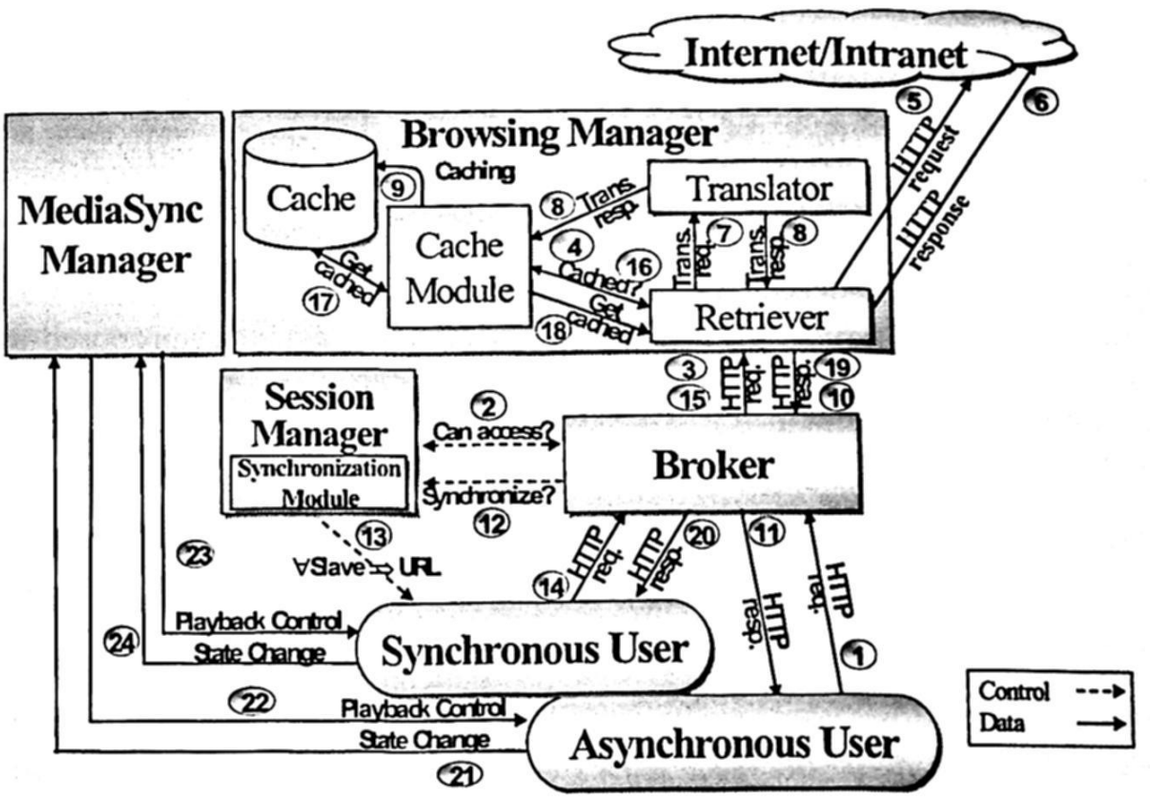


Fig. 3 Synchronization of the browsing and media presentation actions.

4 Synchronization Model of CoLab

In our proposal, we define a CoLab session as a set of users - the session members- engaged in some common browsing activity. In a CoLab session, one or more co-browsing "workgroups," composed of one or more session members, can exist at the same time. During the lifetime of a session, these workgroups can be dynamically created and destroyed. Two workgroups can be merged into a single one, and a single workgroup can be split into two different workgroups, all that under the initiative of the users.

Synchronization Dependency Tree (SDT)

In order to represent the organization of workgroups in a CoLab session, we have chosen to use a data structure called SDT. A typical SDT is shown in Fig. 4a.

1) **Definition 1:** A SDT is a tree structure where nodes represent the users belonging to a single workgroup, and arcs represent the synchronization relations currently existing among them. An arc oriented from node A to node B, where B is the son of A, characterizes the fact that the browsing activities of user B are currently synchronized to those of user A.
2) **Definition 2:** A single user is called an "asynchronous user" if the node representing him in an SDT is the root node (user A in Fig. 4a). This means that this user can freely decide his browsing activities. On the other hand, a single user is called a "synchronous user" if the node that represents him in an SDT belongs to a branch or leaf of the SDT. In this case, all the browsing activities of this user are synchronized to those of the user at the root of the SDT he belongs to (users B, C, and D in Fig. 4a).

The tree structure is quite suitable for representing the organization of workgroups in CoLab since: 1) a single user can get synchronized with only one user and 2) several users can be synchronized at the same time with the same user. This is a natural constraint due to the fact that, if we allow creating cross synchronization relations, we will eventually have conflicts between the interests of two or more users having control of the browsing activity.

As we previously said, an SDT is a dynamic structure since the proposed model allows the dynamic creation and release of synchronization relations among connected users. The creation of a synchronization relation leads to binding the Web browsing of a given user to that of another user. Synchronization relations are created and released by using some predefined synchronization primitives. We can understand this approach as an extension of a classical floor control mechanism, where, in the presence of a synchronization relation, the synchronous user looses his floor in favor of the user he gets synchronized with.

At any given moment during a session, depending on the synchronization relations created and released among the connected users, there can exist different numbers of SDTs. This is called the SDT cardinality and represented by |SDT|. This notion is presented in Fig. 4b. As can be clearly seen, we present here three possible synchronization scenarios for users belonging to a session. In the first case, there exists only one synchronization relation, where user E is currently synchronized with user C, while the other users are asynchronous, so $|SDT| = 4$. In the second case, two

new synchronization relations have been created in such a way that now |SDT| = 2. Finally, in the third case, a new synchronization relation has been created, and two others have been released, taking us to a scenario where |SDT| = 3.

The minimal SDT cardinality of a session is 1 when all the users belong to the same SDT, and the maximal cardinality is equal to the number of connected users when all of them are asynchronous, each one representing a single SDT.

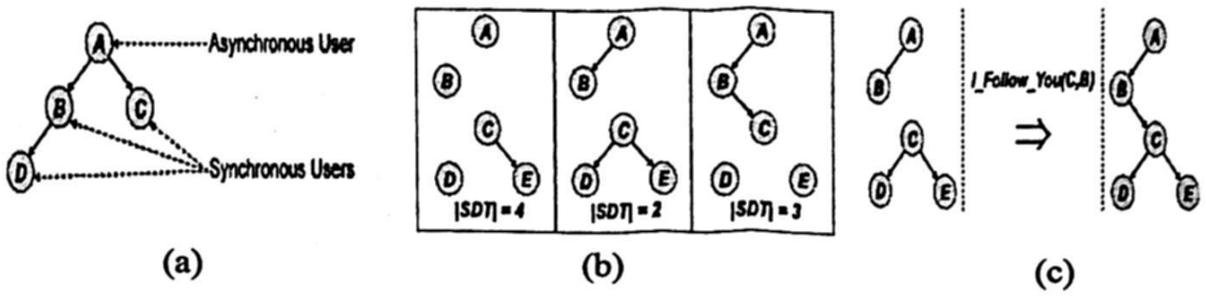


Fig. 4 (a) Basic notion of SDT, (b) SDT configuration scenarios, (c) "I_Follow_You" synchronization primitive.

Synchronization Primitives

CoLab proposes two main synchronization primitives allowing the creation of synchronization relations between users, namely: 1) "I_Follow_You" and 2) "You_Follow_Me". In order to avoid anarchical behaviors, the creation of synchronization relations is subject to an authorization protocol. The "I_Follow_You" primitive provides the user with the possibility of requesting another user his authorization to get synchronized with him. On the other hand, the "You_Follow_Me" primitive provides a user with the possibility of inviting another user to get synchronized with him. Given that a single SDT node may have several children, the "You_Follow_Me" primitive can be applied to a single user as well as to a set of users. As previously stated, whenever either of these two primitives is applied, an authorization protocol is started. The user whom the proposal was sent to is asked whether he wants to accept it. If he accepts, the new synchronization relation is created, and the SDTs of the concerned users are merged. Otherwise, no modification is made. Synchronization relations are released by using the "I_Leave" primitive, which is unconditional: any user involved in a synchronization relation can request it, and it will always succeed. The result of the use of this primitive is that the SDT to which the concerned users belong is split into two single SDTs. Fig. 4c illustrates the session state before (left) and after (right) the use of the "I_Follow_You" primitive. In the left side of the figure, we can see that |SDT| = 2, where users A and C are asynchronous users, user B is synchronized with user A, and users D and E are synchronized with user C. After the use of the "I_Follow_You" primitive from C to B, both SDTs are merged and become a single SDT whose root is user A, so since that moment the browsing activities of all the users of the session will be synchronized with those of user A.

In Fig. 5a, we use an extended state machine-style notation in order to illustrate the general behavior of the synchronization process using the "I_Follow_You" primitive from the point of view of user i. The notation "j!message" means sending

the message "message" to user *j*, and the notation "j?message" means the receival of message "message" from user *j*. In this figure, the two main states in which user *i* can be are "async(),," when the user is working asynchronously, and "sync(),," when the user is synchronized with another user. When user *i* is in the "async()" state, he can use the "I_Follow_You" primitive on user *j*. The preconditions to be able to apply this primitive are: 1) user *i* is asynchronous, and 2) the tree structure is respected. Then, the system passes to an intermediary state where the invitation is expressed to the target user and keeps waiting for an answer to the request: an acceptance, a refusal, or an abort. If an abort or a refusal is expressed, user *i* gets back to the "async()" state, otherwise, the synchronization relation is created, leading, therefore, user *i* passing to the "sync()" state. The behavior of the "You_Follow_Me" primitive is symmetric to that of the "I_Follow_You" primitive, so it will not be presented here. The proposed synchronization model gives CoLab the possibility of supporting the "divide to conquer" concept. The members of a CoLab session are organized into workgroups. Besides, CoLab supports three different organizational structures (based on [1]).

1) Centralized organization, where decisions are made only at the level of the session as a whole. It is more adapted to co-browsing sessions having a leader whose browsing actions must imperatively be followed by the other members (for instance, when a teacher presents a Web-based lecture to a group of students).
2) Decentralized organization composed of different workgroups, where decisions are made independently in each workgroup.
3) Temporarily decentralized organization, which starts out with a decentralized structure and later reintegrates. It is more adapted to co-browsing sessions where the members can browse independently in order to reach the objectives more quickly (for instance, during a collaborative information retrieval), and whenever they decide, they can get their browsing activities synchronized.

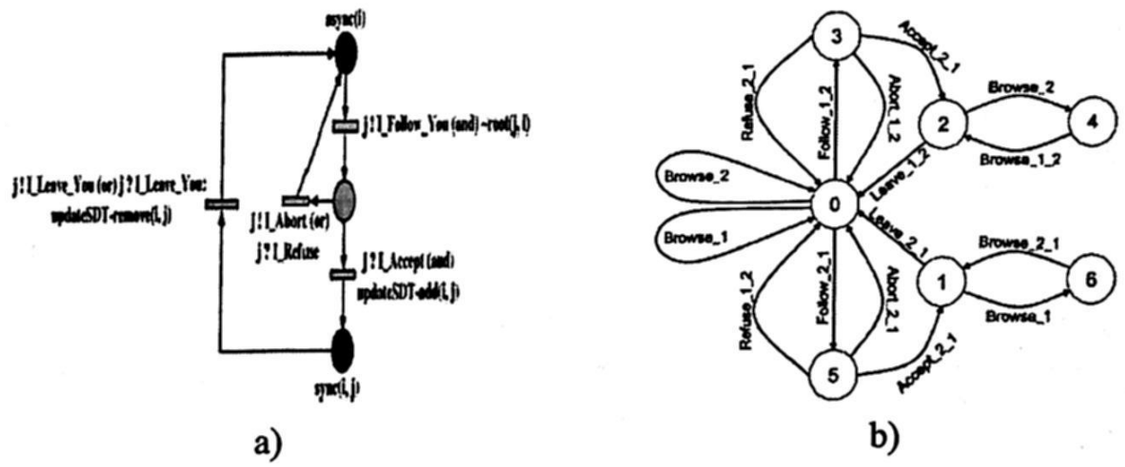


Fig. 5 (a) Basic notion of SDT, (b) SDT configuration scenarios

Synchronization Model Verification

In order to check the consistency of the use of the proposed synchronization primitives, we have formalized them by using Petri nets. Then, we generated some co-browsing scenarios and verified that under any circumstance the complete model is consistent.

As a first step, we defined a set of components representing each of the possible behaviors dealing with the creation or release of synchronization relations, as well as the synchronization of the browsing activities executed by the users.Then, we designed a "TCL" script to generate the Petri net and its initial marking, and we used the software tool "TINA" [8] to get the global reachability graph and the tool "CADP" [9] to obtain an abstract view of this reachability graph (a quotient automaton derived from the reachability graph that features only the synchronization primitives; this automaton is observationally equivalent to the reachability graph, see [10] for details). Fig. 5b shows the complete quotient automaton for a session with two users.

Other results are available for more users (up to five users, due to the classical state space explosion problem) but are not presented here. As can be clearly seen, the connected users can be in either asynchronous or synchronous state, and the browsing activity synchronization behavior is consistent with the current synchronization state of the users.

The state 0 represents that both users are asynchronous: as a consequence, anyone can browse independently without producing any influence in the browsing activity of any other user (transitions "Browse_1" and "Browse_2"). If, for example, user 1 decides to get synchronized with user 2 (transition "Follow_1_2"), the automaton passes to intermediary state 3 waiting for an authorization, abort, or cancel action. Whenever the creation of the synchronization relation is accepted (transition "Accept_2_1"), the automaton passes to state 2, where whenever user 2 executes a browsing action, user 1 is forced to execute exactly the same browsing action (transition "Browse_2" followed of transition "Browse_1_2"). The part of the "Follow_2_1" is symmetric to the "Follow_1_2," so it will not be explained. We have analyzed several scenarios similar to the one presented in Fig. 5b, and we have been able to formally verify that the synchronization model is fully consistent.

5 Related Works

In [11], an adaptation from the technology of unconstrained distributed collaborative editors to develop unconstrained collaborative Web browsing is proposed. However, the effective collaboration is dependent on the awareness of context and group activity. A design of collaborative filtering service platform is described in [12].

The platform provides primitive functions for collaborative filtering that utilizes correlation of user profiles. Basic and extension functions in collaborative filtering especially in the context of distributed environment are discussed.The design of platform is fairly generalized, and it can be realized both in a centralized and peer-to-peer fashion. Furthermore, a load balancing mechanism of the platform is presented.

In [13], new methods for scape-oriented browsing, such as see-through anchors, parallel navigation, and peripheral scape presentations are presented.

A prototype system based in these methods has been designed and implemented. The system offers continuous browsing and navigation to users. A content and device management method for multiple contents browsing with multiple devices is proposed in [14]. Two concepts are introduced: (1) Content Management Description which is used to determine what content to distribute to the device; and (2) Device

Management Description which is used to determine the current status of devices available for browsing content.

This method is expected to achieve effective browsing of contents with multiple devices in users' preferred styles. In [15], a page partitioning method for collaborative browsing is proposed. This method divides a web page into multiple components and each is distributed to a different device. Furthermore, a collaborative web browsing system in which users can search and browse their target information by discussing and watching partial pages displayed on multiple devices is developed.

The closer work is proposed in [16]. Here, a service-oriented architecture for the development of advanced tools for generic service construction and composition is presented. This architecture includes technology-neutral protocols for service instantiation and management with an attempt to encourage development of corresponding tool support. Under this approach, both client side and server side are unified, and GUI services are explicitly modeled; service containers are distinguished from ordinary services that govern service management tasks.

In [17], a collaborative navigation tool called z9 is presented. This tool has as main purpose to selectively transfer anonymous navigation information among a group of users, based on identified user similarities. These similarities are the starting point for the presentation of exploration paths that potentially lead to relevant information. The mains aspects of the development of an awareness tool based on an information-oriented coordination model for synchronous collaboration sessions are proposed in [18]. This tool is supported by an adaptive layered architecture which is based on collaborative extensions of Java language, Java 3D and XML possibilities in terms of data structuring. The application field is related to the execution of a project review for the distributed collaborative design, which is applied to a spatial-domain scenario.

Finally, a Web-based multimedia learning system with automatically generated browsing structures such as hierarchical tables of contents, index and hyperlink is presented in [19]. A pilot study was conducted to evaluate the effectiveness of the system in supporting learning.

6 Conclusions

In this paper, we have defined a general-purpose proxy-based collaborative Web browsing system called CoLab, which is based on a service-oriented architecture, allowing co-browsing by means of a set of operations described as Web Services. We claim that this system gives the users a great flexibility for establishing collaboration relations while browsing, creating in this way an environment where collaboration is greatly facilitated. Our model meets most of the basic requirements for a system aimed at supporting generic synchronous co-browsing applications.

References

1. N. Siggelkow and D. Levinthal, "Temporarily divide to conquer: Centralized, decentralized," Org. Sci., vol. 14, no. 6, pp. 650–669, Nov. 2003.

2. G. J. Hoyos-Rivera, R. L. Gomes, and J. P. Courtiat, "A flexible architecture for collaborative browsing". In Proceedings of the 11th IEEE WetICE, Workshop Web-Based Infrastructures and Coordination Architectures Collaborative Enterprises, Pittsburgh, PA, 2002, pp. 164–169.
3. Bellwood Tom, Clément Luc, Ehnebuske David, Hately Andrew, Hondo Maryann, Husband Yin Leng, Januszewski Karsten.. UDDI Version 3.0 Published Specification. July 19, 2002.
4. Steve Vinoski. Integration with Web Services. IEEE Internet Computing. November- December 2003 pp 75-77.
5. Matthew J. Duftler, Paul Fremantle. Java APIs for WSDL (JWSDL) JSR110: JWSDL Final Release Version 1.0. IBM Corporation March 21, 2003
6. R. L. Gomes, G. J. Hoyos-Rivera, and J. P. Courtiat, "Loosely-coupled integration of CSCW systems". In Proceedings of the 5th IFIP Int. Conf. DAIS, Athens, Greece, Jun. 2005, vol. 3543, pp. 38–49.
7. G. J. Hoyos-Rivera, R. L. Gomes, J. P. Courtiat, and R. Willrich, "Collaborative Web browsing tool supporting audio/video interactive presentation". In Proceedings of the IEEE 14th Int. Workshops WetICE, Linkoping, Sweden, 2005, pp. 78–83.
8. B. Berthomieu, P.-O. Ribet, and F. Vernadat, "The tool TINA-construction of abstract state spaces for petri nets and time petri nets," Int. J. Production Res., vol. 42, no. 14, pp. 2741–2756, 2004.
9. Construction and Analysis of Distributed Processes. (2006). [Online]. Available: http://www.inrialpes.fr/vasy/cadp/
10. R. Milner, A Calculus of Communicating Systems, vol. 92. New York: Springer-Verlag, 1980.
11. Maria Aneiros, Vladimir Estivill-Castro. "Foundations of Unconstrained Collaborative Web Browsing with Awareness". In Proceedings of the IEEE/WIC International Conference on Web Intelligence (WI'03). IEEE Press. 2003.
12. Toshio Oka, Hiroyuki Morikawa, Tomonori Aoyama. "Vineyard: A Collaborative Filtering Service Platform in Distributed Environment". In Proceedings of the 2004 International Symposium on Applications and the Internet Workshops. IEEE Press. 2004.
13. Hiroya Tanaka, Katsumi Tanaka. "WebWalker: Scape-Oriented Web Browsing". In Proceedings of the 12th International Conference on Informatics Research for Development of Knowledge Society Infrastructure. IEEE Press. 2004.
14. Yuhei Akahoshi, Yutaka Kidawara, Katsumi Tanaka. "A Content and Device Management Method for Multiple Contents Browsing with Multiple Devices". In Proceedings of the 21st International Conference on Data Engineering. IEEE Press. 2004.
15. Takuya Maekawa, Takahiro Hara, Shojiro Nishio. "A Collaborative Web Browsing System for Multiple Mobile Users". In Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications. IEEE Press. 2006.
16. Jing-Ying Chen. "Architecting a Service-Oriented Collaborative Web". In Proceedings of the Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services. IEEE Press. 2006
17. Andre Rodrigues da Silva, Vera Lucia Strube de Lima. "z9: An Alternative Approach to Collaborative Navigation". In Proceedings of the Fourth Latin American Web Congress. IEEE Press. 2006
18. L. M. Rodríguez Peralta, A. M. Gonçalves Silva. "A Model-based Awareness Approach for Synchronous Collaborative Sessions on the Web". In Proceedings of the Fourth Latin American Web Congress. IEEE Press. 2006.
19. Ming Lin, Jinwei Cao, Christopher B.R. Diller, Jay F. Nunamaker Jr. "Learning By Browsing: a Web-based multimedia browsing system for learning". In Proceedings of the 39th Hawaii International Conference on System Sciences. IEEE Press. 2006.

Author Index
Índice de autores

Editorial Board of the Volume

Comité editorial de volumen

Additional Reviewers

Árbitros adicionales

Impreso en los Talleres Gráficos
de la Dirección de Publicaciones
del Instituto Politécnico Nacional
Tresguerras 27, Centro Histórico, México, D.F.
Mayo de 2007.
Printing 500 / Edición 500 ejemplares.

This volume contains 26 carefully selected papers by 79 authors from 9 countries: Argentila, Brazil, Chile, Cuba, France, Ireland, Japan, Mexico and Spain. These papers present the most recent developments in a range of areas related to computer science and engineering. The papers are arranged into 9 thematic fields:

- Knowledge Acquisition and Neural Networks
- Fuzzy Logic
- Optimization
- Image Processing
- Natural Language Processing
- Agents
- Time Series
- Robotics and Control
- Computer Networks and Web Services

The volume will be useful for researchers, students, and general public interested in the corresponding areas of computer science and engineering.

ISSN: 1870-4069
www.cic.ipn.mx/rcs
www.ipn.mx

SEP
SECRETARÍA DE EDUCACIÓN PÚBLICA

INSTITUTO POLITÉCNICO NACIONAL
"La Técnica al Servicio de la Patria"