ADVANCES IN INTELLIGENT AND *SOFT COMPUTING* 71

Yves Demazeau et al. (Eds.)

Trends in Practical Applications of Agents and Multiagent Systems

Springer

Advances in Intelligent and Soft Computing 71

Editor-in-Chief: J. Kacprzyk

Advances in Intelligent and Soft Computing

Editor-in-Chief

Prof. Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
ul. Newelska 6
01-447 Warsaw
Poland
E-mail: kacprzyk@ibspan.waw.pl

Further volumes of this series can be found on our homepage: springer.com

Vol. 56. H. Wang, Y. Shen,
T. Huang, Z. Zeng (Eds.)
The Sixth International Symposium on Neural Networks (ISNN 2009), 2009
ISBN 978-3-642-01215-0

Vol. 57. M. Kurzynski,
M. Wozniak (Eds.)
Computer Recognition Systems 3, 2009
ISBN 978-3-540-93904-7

Vol. 58. J. Mehnen, A. Tiwari,
M. Köppen, A. Saad (Eds.)
Applications of Soft Computing, 2009
ISBN 978-3-540-89618-0

Vol. 59. K.A. Cyran,
S. Kozielski, J.F. Peters,
U. Stańczyk, A. Wakulicz-Deja (Eds.)
Man-Machine Interactions, 2009
ISBN 978-3-642-00562-6

Vol. 60. Z.S. Hippe,
J.L. Kulikowski (Eds.)
Human-Computer Systems Interaction, 2009
ISBN 978-3-642-03201-1

Vol. 61. W. Yu, E.N. Sanchez (Eds.)
Advances in Computational Intelligence, 2009
ISBN 978-3-642-03155-7

Vol. 62. B. Cao,
T.-F. Li, C.-Y. Zhang (Eds.)
Fuzzy Information and Engineering Volume 2, 2009
ISBN 978-3-642-03663-7

Vol. 63. Á. Herrero, P. Gastaldo,
R. Zunino, E. Corchado (Eds.)
Computational Intelligence in Security for Information Systems, 2009
ISBN 978-3-642-04090-0

Vol. 64. E. Tkacz, A. Kapczynski (Eds.)
Internet – Technical Development and Applications, 2009
ISBN 978-3-642-05018-3

Vol. 65. E. Kącki, M. Rudnicki,
J. Stempczyńska (Eds.)
Computers in Medical Activity, 2009
ISBN 978-3-642-04461-8

Vol. 66. G.Q. Huang,
K.L. Mak, P.G. Maropoulos (Eds.)
Proceedings of the 6th CIRP-Sponsored International Conference on Digital Enterprise Technology, 2009
ISBN 978-3-642-10429-9

Vol. 67. V. Snášel, P.S. Szczepaniak,
A. Abraham, J. Kacprzyk (Eds.)
Advances in Intelligent Web Mastering - 2, 2010
ISBN 978-3-642-10686-6

Vol. 68. V.-N. Huynh, Y. Nakamori,
J. Lawry, M. Inuiguchi (Eds.)
Integrated Uncertainty Management and Applications, 2010
ISBN 978-3-642-11959-0

Vol. 69. XXX

Vol. 70. Y. Demazeau, F. Dignum,
J.M. Corchado, J. Bajo (Eds.)
Advances in Practical Applications of Agents and Multiagent Systems, 2010
ISBN 978-3-642-12383-2

Vol. 71 Y. Demazeau, F. Dignum,
J.M. Corchado, J. Bajo, R. Corchuelo,
E. Corchado, F. Fernández-Riverola,
V.J. Julián, P. Pawlewski, A. Campbell (Eds.)
Trends in Practical Applications of Agents and Multiagent Systems, 2010
ISBN 978-3-642-12432-7

Yves Demazeau, Frank Dignum,
Juan M. Corchado, Javier Bajo,
Rafael Corchuelo, Emilio Corchado,
Florentino Fernández-Riverola,
Vicente J. Julián, Pawel Pawlewski, and
Andrew Campbell (Eds.)

Trends in Practical Applications of Agents and Multiagent Systems

8th International Conference on Practical Applications of Agents and Multiagent Systems

Springer

Editors

Yves Demazeau
Laboratoire d'Informatique de Grenoble
Centre National de la Recherche
Scientifique, Maison Jean Kuntzmann
110 av. de la Chimie
F-38041 Grenoble, France
E-mail: yves.demazeau@imag.fr

Frank Dignum
Department of Information and
Computing Sciences
Universiteit Utrecht Centrumgebouw Noord,
office A117, Padualaan 14, De Uitho
3584CH Utrecht, The Netherlands
E-mail: dignum@cs.uu.nl

Juan M. Corchado
Departamento de Informática y Automática
Facultad de Ciencias Universidad
de Salamanca Plaza de la Merced
S/N 37008 Salamanca, Spain
E-mail: corchado@usal.es

Javier Bajo
Escuela Universitaria de Informática
Universidad Pontificia de Salamanca,
Compañía 5, 37002 Salamanca, Spain
E-mail: jbajope@upsa.es

Rafael Corchuelo
ETSI Informática Universidad
de Sevilla Avda. Reina Mercedes
s/n 41012 Sevilla, Spain
E-mail: corchu@us.es

Emilio Corchado
Departamento de Informática
y Automática Facultad de Ciencias
Universidad de Salamanca Plaza de la
Merced S/N 37008 Salamanca, Spain
E-mail: escorchado@ubu.es

Florentino Fernández-Riverola
Escuela Superior de Ingeniería
Informática, Edificio Politécnico,
Despacho 408, Campus Universitario
As Lagoas s/n, 32004 Ourense, Spain
E-mail: riverola@ei.uvigo.es

Vicente Julián
Departamento de Sistemas Informáticos
y Computación Universidad Politécnica
de Valencia, Camino de Vera s/n
46022 Valencia, Spain
E-mail: vinglada@dsic.upv.es

Pawel Pawlewski
Department of Computing and Management
Poznan University of Technology
Strzelecka Str. 11, 60965 Poznan, Poland
E-mail: pawel.pawlewski@put.poznan.pl

Andrew Campbell
Department of Computer Science
Dartmouth College 6211 Sudikoff Laboratory
Hanover, NH 03755-3510, USA
E-mail: campbell@cs.dartmouth.edu

ISBN 978-3-642-12432-7 e-ISBN 978-3-642-12433-4

DOI 10.1007/978-3-642-12433-4

Advances in Intelligent and Soft Computing ISSN 1867-5662

Library of Congress Control Number: 2010923872

Typeset & Cover Design: Scientific Publishing Services Pvt. Ltd., Chennai, India.

Printed on acid-free paper
5 4 3 2 1 0
springer.com

Preface

New trends and strategies on Agents and Multi-Agent Systems have recently appeared and many effective applications of this technology are now deployed. An international forum to present and discuss the latest scientific trends and strategies on practical applications of Agents and Multi-Agent Systems, to assess the impact of the approach, and to facilitate technology transfer, has become a necessity.

PAAMS, the International Conference on Practical Applications of Agents and Multi-Agent Systems is an international yearly forum to present, to discuss, and to disseminate the latest developments and the most important outcomes related to real-world applications. It provides a unique opportunity to bring multi-disciplinary experts, academics and practitioners together to exchange their experience in the development of Agents and Multi-Agent Systems.

This volume presents the papers that have been accepted for the 2010 edition in the Special Sessions and Workshops. PAAMS 2010 Special Sessions and Workshops are a very useful tool in order to complement the regular program with new or emerging topics of particular interest to the participating community. Special Sessions and Workshops that emphasize on multi-disciplinary and transversal aspects, as well as cutting-edge topics were especially encouraged and welcomed.

PAAMS 2010 included a total of 7 special sessions and 2 workshops: Special sessions on Adaptive Multi-agent Systems, Multi-agent systems for Health Care and Bioinformatics, Multi-agent systems for Ambient Intelligence, Multi-Agent Systems for Manufacturing and Supply Chains, Enterprise Application and Information Integration, Software Agents in Knowledge Management and Bio-inspired and Multi-Agent Systems: Applications to Languages. The workshop on Artificial Intelligence and Distributed Computing included sessions on Sensing Systems and Intelligence and Ageing Well in the Knowledge Society. Finally, the workshop on on Systems, Man, & Cybernetics: SMC-Workshop.- IEEE-SPANISH CHAPTER provided an interesting opportunity to present and discuss the latest theoretical advances related to agents in the IEEE Spanish Chapter.

We would like to thank all the contributing authors, as well as Workshop and special session organizers and the members of the Program Committees and the

Organizing Committees for their hard and highly valuable work. Their work has helped to contribute to the success of the PAAMS 2010 event. Thanks for your help, PAAMS 2010 wouldn't exist without your contribution.

Yves Demazeau	Juan Manuel Corchado
Frank Dignum	Javier Bajo
PAAMS 2010 Program Co-chairs	PAAMS 2010 Organizing Co-chairs

Organization

General Co-chairs

Yves Demazeau	Centre National de la Recherche Scientifique (France)
Frank Dignum	Utrecht University (The Netherlands)
Juan M. Corchado	University of Salamanca (Spain)
Javier Bajo	Pontifical University of Salamanca (Spain)

Program Committee

Yves Demazeau (Co-chairman)	Centre National de la Recherche Scientifique (France)
Frank Dignum (Co-chairman)	Utrecht University (The Netherlands)
Francesco Amigoni	Politecnico di Milano (Italy)
Luis Antunes	University of Lisbon (Portugal)
Olivier Boissier	Ecole Nationale Superieure des Mines de Saint Etienne (France)
Magnus Boman	Royal Institute of Technology (Sweden)
Juan A. Botía	University of Murcia (Spain)
Vicente Botti	Polytechnic University of Valencia (Spain)
Jeffrey Bradshaw	Florida Institute for Human and Machine Cognition (USA)
Bernard Burg	Panasonic Ltd (USA)
Valerie Camps	University Paul Sabatier (France)
Longbing Cao	University of Technology Sydney (Australia)
Pierre Chevaillier	University of Brest (France)
Owen Cliffe	University of Bath (UK)
Juan M. Corchado	University of Salamanca (Spain)
Rafael Corchuelo	University of Sevilla (Spain)
Keith Decker	University of Delaware (USA)

Jörg Müller	Clausthal University of Technology (Germany)
Jean-Pierre Muller	CIRAD-TERA-REV-GREEN (France)
James Odell	James Odell Consultancy (USA)
Eugenio Oliveira	University of Porto (Portugal)
Andrea Omicini	University of Bologna (Italy)
Sascha Ossowski	University of Rey Juan Carlos (Spain)
Julian Padget	University of Bath (UK)
Van Parunak	New Vectors (USA)
Juan Pavón	Universidad Complutense de Madrid (Spain)
Michal Pechoucek	Czech Technical University in Prague (Czech Republic)
Paolo Petta	University of Vienna (Austria)
Jeremy Pitt	Imperial College of London (UK)
Antonio Rocha Costa	Universidade Católica de Pelotas (Brazil)
Juan A. Rodríguez Aguilar	IIIA-CSIC (Spain)
Fabrice Saffre	British Telecom (UK)
Jaime Sichman	University of Sao Paulo (Brazil)
Kostas Stathis	Royal Holloway University of London (UK)
John Thangarajah	RMIT (Australia)
Paolo Torroni	University of Bologna (Italy)
Rainer Unland	University of Duisburg (Germany)
Domenico Ursino	University of Reggio Calabria (Italy)
Birna van Riemsdijk	University of Delft (The Netherlands)
Javier Vázquez Salceda	Polytechnic University of Catalogna (Spain)
Jacques Verriet	Embedded Systems Institute (The Netherlands)
Danny Weyns	Catholic University of Leuven (Belgium)
Niek Wijngaards	Thales, D-CIS lab (The Netherlands)
Michael Winikoff	University of Otago (New Zealand)
Franco Zambonelli	University of Modena (Italy)

Special Session on Adaptive Multiagent Systems Committee

Stella Heras (Co-chairman) - Universidad Politécnica de Valencia
Martí Navarro (Co-chairman) - Universidad Politécnica de Valencia
Vicente Julián (Co-chairman) - Universidad Politécnica de Valencia

Special Session on Multiagent Systems for Health Care and Bioinformatics Committee

Florentino Fernández-Riverola (Co-chairman) - University of Vigo
Paulo Novais (Co-chairman) - University of Minho
Fernando Díaz - University of Valladolid
Daniel González-Peña - University of Vigo
Miguel Rocha - University of Minho
Juan Francisco de Paz - University of Salamanca

Special Session on Multiagent Systems for Ambient Intelligence Committee

Dante I. Tapia (Chairman) - University of Salamanca
Miguel A. Valero - Polytecnic University of Madrid
Luis Alonso - University of Salamanca
Diego López de Ipiña - University of Deusto
Oscar García - University of Valladolid
Juan A. Botía - University of Murcia
Ricardo S. Alonso - University of Salamanca
Pau Giner - Polytechnic University of Valencia

Special Session on Multi-Agent Systems for Manufacturing and Supply Chain Committee

Pawel Pawlewski (Co-chairman) - Poznan University of Technology
Paulina Golinska (Co-chairman) - Poznan University of Technology

Special Session on Enterprise Application and Information Integration Committee

Rafael Corchuelo Gil (Co-chairman) - University of Sevilla
Rafael Z. Frantz (Co-chairman) - UNIJUI

Special Session on Software Agents in Knowledge Management Committee

Aurora Vizcaino (Chairman) - University of Castilla La Mancha
Ricardo Imbert - Politechnical University of Madrid
Javier Portillo - University of Castilla La Mancha
Alberto Fernández - University Rey Juan Carlos

Special Session on Bio-Inspired and Multi-Agents Systems: Applications to Languages Committee

Gemma Bel Enguix (Co-chairman) - GRLMC - Rovira i Virgili University
Alfonso Ortega de la Puente (Co-chairman) - Universidad Autónoma de Madrid

Workshop on Artificial Intelligence and Distributed Systems Committee

Andrew Campbell (Co-chairman) - Dartmouth College
Sara Rodríguez (Co-chairman) - Universidad de Salamanca
Juan F. de Paz - University of Salamanca
Vidal Alonso - Universidad Pontificia de Salamanca
Vivian López - Universidad de Salamanca
Carlos Carrascosa - Universidad Politécnica de Valencia
Emilio Corchado - Universidad de Burgos
Fernando Díaz - Universidad de Valladolid
Álvaro Herrero - Universidad de Burgos
Angélica Gonzalez - Universidad de Salamanca
Florentino Fdez-Riverola - Universidad de Vigo
Celia Gutierrez - Universidad Complutense de Madrid
Julian Padget - Univ. of Bath
Javier Arroyo - Universidad Complutense de Madrid
María Moreno - Universidad de Salamanca
Vicente Julian - Universidad Politécnica de Valencia
Belén Pérez-Lancho - Universidad de Salamanca
Javier Bajo - Universidad Pontificia de Salamanca
Oscar Gil Gonzalo - Universidad de Salamanca
Ulises Cortés - Technical University of Catalonia
Roberta Annicchiarico - Fondazione Santa Lucia
Femida Gwadry-Sridhar - U. Western Ontario
Patty Koskova - City University
Magí Lluch - Microart
Mike Luck - King's College
José M. Molina - Universidad Carlos III de Madrid
Antonio Moreno - U. Rovira i Virgili
Javier Vázquez-Salceda - Technical Univ. of Catalonia

Workshop on System, Man, & Cybernetics: SMC-Workshop.-IEEE.-SPANISH CHAPTER Committee

Emilio Corchado (Chairman) - Universidad de Salamanca
Manuel Graña - University of Basque Country
José Manuel Molina - University Carlos III of Madrid
Álvaro Herrero - University of Burgos

Richard Duro - University of Coruña
Juan M. Corchado - University of Salamanca
Ramón Rizo - University of Alicante
Juan Pavón - University Complutense of Madrid
Vicent Botti - Polytechnical University of Valencia
Bruno Baruque - University of Burgos
Javier Sedano - University of Burgos
Lourdes Sáiz Barcena - University of Burgos
Roberto Alcalde Delgado - University of Burgos
Sara Rodríguez - University of Salamanca

Organizing Committee

Juan M. Corchado (Co-chairman)	University of Salamanca (Spain)
Javier Bajo (Co-chairman)	Pontifical University of Salamanca (Spain)
Juan F. De Paz	University of Salamanca (Spain)
Sara Rodríguez	University of Salamanca (Spain)
Dante I. Tapia	University of Salamanca (Spain)
M. Dolores Muñoz Vicente	University of Salamanca (Spain)

Contents

Special Session on Adaptive Multiagent Systems

Special Session on Multiagent Systems for Health Care and Bioinformatics

Special Session on Multiagent Systems for Ambient Intelligence

Special Session on Multi-Agent Systems for Manufacturing and Supply Chain

Special Session on Enterprise Application and Information Integration

Workshop on Artificial Intelligence and Distributed Systems

Workshop on Systems, Man, & Cybernetics: SMC-Workshop.- IEEE.- SPANISH CHAPTER

Use of Agents to Realize a Federated Searching of Learning Objects

Jaime Muñoz-Arteaga, Edgar A. Calvillo-Moreno, Carlos A. Ochoa-Zezzatti, René Santaolaya-Salgado, and Fco. Álvarez-Rodríguez

Abstract. A traditional search applied in a federation of learning objects repositories consumes a lot of resources in time, large numbers of queries and registers. The main improvement proposed in this work is the insertion of agents approach in order to have pertinent results with less consuming resources and covering the expectations of user. An advantage to use the agent approach is possible to apply semantic searching of learning objects.

1 Introduction

Learning objects is considered as educational resources that can be employed in technology support learning. They are a digital pieces of knowledge to put together in order to form courses on line. Learning object are generally saved and accessed in repositories which offer a series of services such as display, update, and maintain this kind of objects. In addition, this work consider the repositories could be connected and distributed in different academic institutions such as a federation of learning object repositories, where a federated search could be applied in any point of federation. Hence a teacher can develop a course in a transparent way searching and selecting learning objects from different repositories.

Jaime Muñoz-Arteaga · Edgar A. Calvillo-Moreno · Fco. Álvarez-Rodríguez
Universidad Autónoma de Aguascalientes, Av. Universidad # 940, Aguascalientes, México
e-mail: jmauaa@gmail.com, alancalvillo@yahoo.com, fjalvar@correo.uaa.mx

Carlos A. Ochoa-Zezzatti
Universidad Autónoma de Ciudad Juarez, Henry Dunant, 32310 Cd. Juárez, México
e-mail: alberto.ochoa@uacj.mx

Jaime Muñoz-Arteaga · René Santaolaya-Salgado
CENIDET, Interior Internado Palmira s/n. Col. Palmira C.P. 62490, Cuernavaca, Morelos, México
e-mail: rene@cenidet.edu.mx

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 1–8.
springerlink.com

A traditional search applied in a federation of learning objects repositories consumes a lot of resources in time, large numbers of queries and registers. The main improvement proposed in this work is the insertion of intelligent agents in order to have pertinent results with less consuming resources and covering the expectations of user communities. Additional advantages using the agent approach in a federated search, it is possible to get the learning objects in a semantic way.

For this goal this work is structured in six sections, next section specify some problems to take into account for this kind of repositories. Our proposal is presented in section three with different abstract models given answers to main problems. Section four presents a case study in order to apply and prove our proposal. Finally, our proposal is compared with another related work.

2 Outline Problem

Nowadays a large number of universities are producing their courses in terms of learning objects and saving these objects in their own repositories, these repositories in general support several local queries with different criteria thanks to the information in the metadata. When the universities connect their repositories with other academic institutions in order to reuse the contents and increase the number of courses, at this moment a set of problems start to appear affecting the answer time for every query, for example a research of a small number of learning objects require between 15 to 20 seconds, with this time decrease the productivity of user task.

The problem becomes more complex, when a final user applies a search with a combination of several criteria and he/she requires results with several abstraction levels of representation. The repositories save large collections of objects that can be interconnected with other repositories, hence if the connections rise between the repository the search mechanism become more complex.

Another problem is the diversity of learning management system could require different services to the repositories, examples of these services are the visualization, search, send, evaluation of learning objects required by user. Today the user could prepare and/or attend several coursed on line using different learning Management System (LMS) such as Moodle, Claroline, WebCT, etc… For these LMS, the user require frequently a group of same services such as the search of learning objects, this is a very useful service for all kind of user, but it is managed and programmed very differently in current learning object repositories.

Finally, most repositories lack of conceptual model to specify several aspects to design such as the restrictions and facilities for the communication and interoperability in order to offer better services to user notably the teacher and students [13].

In brief, it is possible identify a list of difficulties for the service of searching of learning objects:

- A transparent communication and interoperability between the repositories
- Carry out searches in a quick and efficient way.
- Complexity of semantic level
- Give efficient visualization of learning object
- Establish a direct connection with different LMS (Learning Management System).
- The results require several abstraction levels of representation.

3 Modeling a Federated Searching of Learning Objects

When a user searches any information using an interactive system in a learning process, it is necessary to get property the user task as well as the feedback to user. The term ‘‘user feedback” is often referred to any answer given by a system to user. We propose the agent approach in order to improve the time for the feedback to user. we suggest the use of web services in order the search of learning objects is viewed as only one service for the federation of repositories (see figure 1); this is transparent for users since they continue to access their courses composed from different learning objects coming from institutions which sharing their academics contents.

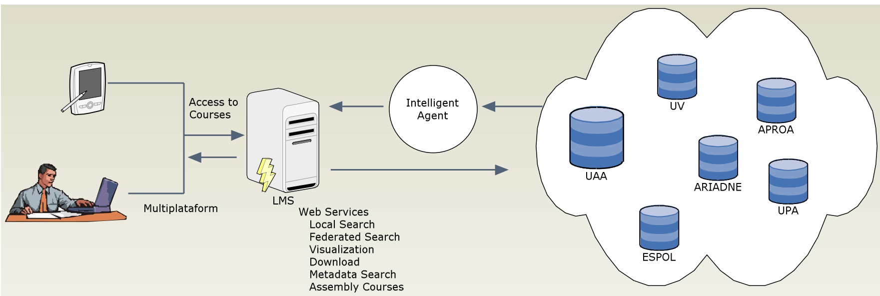


Fig. 1 Federated searching of learning objects

Then, it is possible to define a federated search as a quest applied to the repositories using web services and give as a result a set of learning objects which satisfy the user expectative. All the results of a federate search are given to an intelligent agent in order to answer properly and a short time the user feedback. This information must to be shown in an adequate manner to user and satisfy the queries required by user (see figure1). A good performance of feedback and gather important information, this is one of the goals of intelligent agent in order to send it to the graphical user interface.

In the literature a Multi-Agent System (MAS) is a system composed of multiple interacting intelligent agents Multi-agent systems can be used to solve different problems related with argumentation [2], reputation [9] and negotiation [4] which is difficult for an individual agent or monolithic system to solve. Examples of problems which are appropriate to multi-agent systems research include online trading, disaster response, and modeling social structures. The majority of Intelligent Agents using a Behavioral Model named the Belief-Desire-Intention (BDI) software model (usually referred to simply, but ambiguously, as BDI) is a software model developed for programming intelligent agents. Superficially characterized by the implementation of an agent's beliefs, desires and intentions, it actually uses these concepts to solve a particular problem in agent programming. In essence, it provides a mechanism for separating the activity of selecting a plan (from a plan library) from the execution of currently active plans. Consequently, BDI agents are able to balance the time spent on deliberating about plans (choosing what to do) and executing those plans (doing it). A third activity, creating the plans in the first place (planning), is not within the scope of the model, and is left to the system designer and programmer.

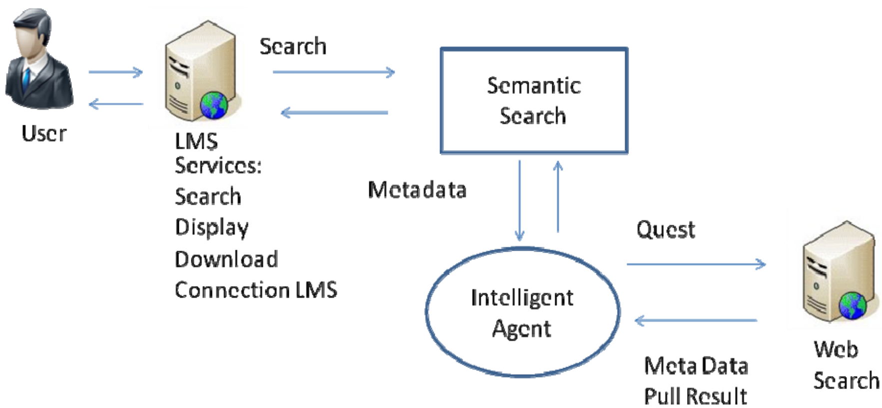


Fig. 2 Search of learning objects using an agent

The figure 2 shows the process necessary to recovery of metadata of learning objects distributed throughout the federation of repositories. This information will be used by the intelligent agents to implant inside of the federated search; all the information in a single server help to have a better performance in time. The searches at semantic level take part key data contained metadata for the time being in the learning objects, like they are name of the object, title, description, and authors. This information is introduced and once carried out this the agent takes the information processes it among the different institutions. This extracted information of the metadata takes inside the web service to form a group of key information related to the learning objects inside the federation. Once the agent has a result containing detailed information, then it is possible to offer in final a user a better precise a search with a minimum of time.

4 Implementation of Model

Current conceptual model of this work had been implemented with some repositories of some Mexican universities, they are sharing their academic contents in order to conform the Mexican learning object repositories composed by the repositories from Universidad Veracruzana (UV), Universidad Politecnica de Aguascalientes (UPA), Cenidet and the Universidad de Aguascalientes (UAA). As it is shown in figure 3 every repository have several learning objects, in particular in the domain of biology, computer and social sciences.

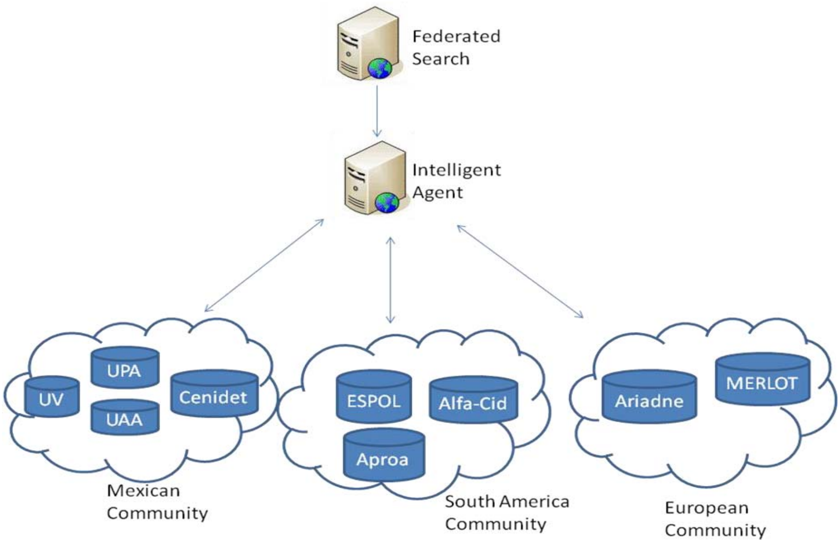


Fig. 3 Architectural model for several federation of learning object repositories

This initial group of Mexican federation of learning object repositories is connected now with other repositories from the south America and European communities. as a result, the search service help to a teacher select and identify different learning objects coming from the universities sharing their academics contents notably in Spanish language. For example at Universidad Autónoma de Aguascalientes it is possible to access to the repository ESPOL in Ecuador which have a large number learning objects and it allows access the European Ariadne repository [4].

Next figure shows an example of a result of a federate search, where a user has required some learning objects in the domain of data structure, this subject is entered by user using a learning management system. Hence, at semantic level, some key data are identified in the metadata of a learning object, such as title, description, authors and a short description. Once carried out this, the agent receives the results in terms of metadata of different learning objects, maybe this objects are under different standard, structure and format. The agent proposed here also assures that every learning object could be under the standard SCORM in order to be accessible to user.

One example is given in the figure 4 showing a list of learning objects in the domain of data structure. Thanks to the procedure accomplished by the agent it is possible that every learning object could be selected, downloaded and visualized by user.

The previous figure shows a learning object listed in figure 4, it is a learning object visualized a content of an Array, this object comes from the repository of Universidad Autónoma de Aguascalientes (UAA) . The agent structures the content of every learning object in terms of five components: presentation, theory, section to put in practice, evaluation and bibliography. With this structure, the teachers could offer to their student a course on line composed by a set of uniform learning objects.

Fig. 4 A List of learning objects as result of a federate search

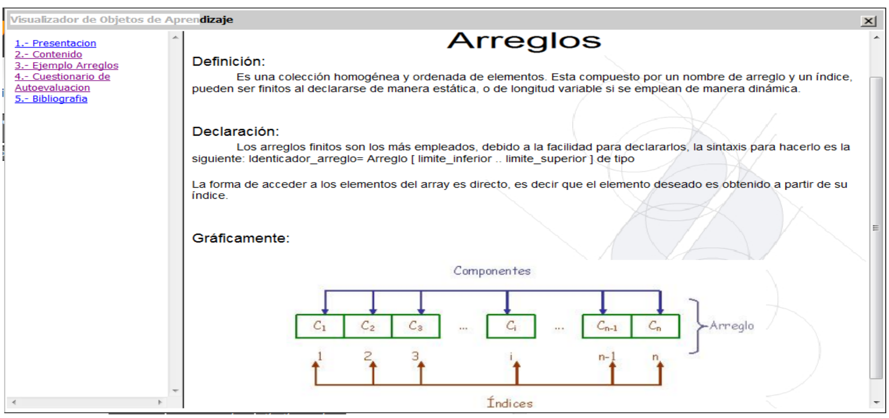


Fig. 5 An array presented as a learning object

5 Related Work

In this section we compare current approach with another related work. We use the following criteria to identify advantages and disadvantages of our research: web services, semantic search, federation of repositories, user of anthologies, us use of agent approach.

Table 1 Some related works wih learning object repositories

Criteria/Repositories	RedoUAA	Ariadne [11]	Espol [13]	Aproa [12]
Web Services	X	X	X	X
Agent approach	X			
Semantic Search	X	X		
Member of a federation	X	X	X	
Use of Ontologies		X		

Previous table shows that Ariadne and the repository called (RedoUAA) of Universidad de Aguascalientes, they cover better the set of criteria, one work use the ontologies and the other one use the agent approach in order to search better any learning object taking into account user context.

5 Conclusions

This paper proposes a model for search learning objects in a federation of repositories using agents. The agent proposed here help to manage the learning objects coming from the search in different repositories and it helps to reduce the time of user feedback. The model is implemented by with a group of learning object repositories from some Mexican universities; then, an application of federated searching of learning objects is shown.

There are several aspects as future work such the semantic search could be based on a multi agents and ontology models, thus it would be possible to provide a set of educational criteria (pedagogical, economic or other) in order to provide a set of results that better accomplished the end-user needs. In the same way, it is necessary a model for a systematic production of intelligent learning objects based on cultural algorithms in the domain of astronomy and nanotechnology.

References

1. Sánchez-Alonso, S., Soto, J., Sicilia, M.A.: Designing Flexible Learning Object Repositories: Balancing Flexibility and Delegation in Ontological Characterizations. In: Harman, K., Koohan, A. (eds.) Learning Objects: standards, metadata, repositories, & LCMs, pp. 221–253. Informing Science press, Santa Rosa (2007), Heras, S., Rebollo, M., Julián, V.: Arguing About Recommendations in Social Networks. In: IAT 2008, pp. 314–317 (2008)
2. Lopez Guzman, C., Garcia Peñalevo, F.: Formation of learning Objects through the use of the metadatos of a digital collection: of Dublin Core to IMS. In: SPDECE 2004, Guadalajara, Spain, October 20-24 (2008)
3. Ochoa-Zezzatti, C., Padilla, A., González, S., Castro, A.: Shikari Hal: Improve a Game Board Base on Cultural Algorithms. IJVR 7(2), 41–46 (2008)

4. Muñoz-Arteaga, J., Ochoa, X., Calvillo-Moreno, E., Vincent, G.: Integration of RE-DOUAA to the Latin American Federation of Repositories of Learning Objects. In: LACLO 2007 (2da. he/she confers Latin American of Objects of Learning), Santiago from Chile, October 22-25 (2007)
5. Barcelo-Valenzuela, M., Alfonso-Sanchez, G., Perez-Soltero, A.: La Web Semantica como apoyo a la gestión del conocimiento y el modelado organizacional. Revista Ingenieria Informatica, Edicion 12 (Abril 2006)
6. Nikraz, M., Caire, G., Bahri, P.A.: A Methodology for the Analysis and Design of Multi-Agent System Using Jade
7. Jacsó, P.: Thoughts About Federated Searching. Information Today 21(9) (October 2004)
8. Conte, R., Paolucci, M., Mir, J.S.: Reputation for Innovating Social Networks. Advances in Complex Systems 11(2), 303–320 (2008)
9. Robert, A.: The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration. Princeton University Press, Princeton (1997)
10. Duval, E., Vidal, P., et al.: Ariadne Knowledge Pool System, version 5, http://www.ariadne-eu.org/index.php (Access date: October 2009)
11. Inostroza, P., Santibañez, F.: Aprendiendo con Objetos de Aprendizaje, http://www.aproa.cl/ (Access date: October 2009)
12. Ochoa, X.: ESPOL, http://www.ariadne-eu.org/index.php (Access date October 2009)

MDD for Virtual Organization Design

Jorge Agüero, Miguel Rebollo, Carlos Carrascosa, and Vicente Julián

Abstract. This paper presents an approach to design a *Virtual Organization* using MDD (Model Driven Development), in a fast and simple way. This approach employs a meta-model of *Virtual Organization* which has concepts and components at a suitable level of abstraction so that it can be implemented on different systems. The user will design the system for different platforms by means of unified agent models (UML-Like). The development process is presented as a set of steps which cover from the model design to the code generation.

1 Introduction

Virtual Organizations (VOs) are a set of individuals and institutions that need to coordinate resources and services across institutional boundaries[3]. Thus, they are open systems formed by the grouping and collaborative of heterogeneous entities. The *Organization* describes the main aspects of a society based on different viewpoints, such as: structure, functionality, norms, interactions, and environment. These societies (organizations) require large levels of interoperability to integrate diverse information systems to share knowledge and collaborate among organizations[3].

Hereby, the organization needs to employ basic sotfware components which support the development of fast and easy composition of distributed applications even in heterogeneous environments systems. Where the components are assembled with little effort into of another applications cooperatively to create flexible dynamic processes. These levels of flexibility and cooperation between different software components is achieved using *Agent-Oriented Software Engineering* (AOSE)[13].

Software engineering based on Multi-Agent Systems (MAS) are a powerful technology with very significant applications in distributed systems and artificial

Jorge Agüero · Miguel Rebollo · Carlos Carrascosa · Vicente Julián
Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia
Camino de Vera S/N 46022 Valencia, Spain
e-mail: {jaguero,mrebollo,carrasco,vinglada}@dsic.upv.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 9–17.
springerlink.com

intelligence [12]. MAS supporting all of these developments requires the creation of platforms of highly heterogeneous agents, where agents work together through different interactions to support complex tasks, in a collaborative and dynamic way. With this in mind, it is suitable for agents to display characteristics such as sociability, autonomy, self-organization, etc. Therefore, it is necessary to create open systems composed of a group of cooperative and heterogeneous agents, which work with local or individual goals and that must fulfill global goals.

A major challenge when designing MAS is to provide efficient tools that can be used by any user (non-expert users). The MDD approach can facilitate and simplify the design process and the quality of agent-based software[14], since it allows the reuse of software and transformation between models. MDD basically proposes the automatic generation of code from the models using the transformations. In other words, using models that have components that are platform independent, and by means of the transformations. Those models are translated into components (or code) that depend on the execution platform, which integrates specific details about the system. Recently some proposals to implement these ideas have been proposed in MAS[1, 10, 11], but none of these proposals focus on organizational design.

This work proposes a development process to obtain VOs model which can be implemented in different platforms applying the basic ideas of MDD. In this way, a non-expert programmer will be facilitated to develop VOs, reducing the gap between design and implementation. So, two conversion models for translate a single model of the VO to two different platforms are also proposed, allowing the feasibility of the proposal to be verified. These target platforms are: THOMAS[1][4] and E-Institutions[2][7].

This paper is structured as follows: a brief summary of relevant works and their problems are discussed in Section 2. Section 3 provides core concepts of VO and MDD. Section 4 shows how to apply the MDD approach to develop VOs. Finally, the conclusions are presented in Section 5.

2 Related Work

This section presents some related contributions with respect to organization modeling in agent-based systems, and discusses some problems. Furthermore, this section explains how, using the MDD approach, these problems can be addressed.

Virtual Organization design. MAS development needs methodologies that allow the design of agent-based software to be optimized. The first methodologies that emerged can be classified as *agent-oriented*, without describing the *organization* explicitly. These systems are generally closed and external agents are prohibited. But in recent times, new *organizational-oriented* methodologies have emerged, which allow (partially) the design of open MAS, leading to the development *heterogeneous* systems. These organizational-oriented methodologies allow external agents to access the system functionality, but the agents are obliged to adhere to the

[1] http://users.dsic.upv.es/grupos/ia/sma/tools/Thomas

[2] http://e-institutions.iiia.csic.es

social norms of the system. Among the most important methodologies which allow the design of *Organizations* are: MOISE[9], OperA[6] and GORMAS[3]. However, these works do not include all the phases for develop the open MAS, mainly the latter phases, in which the *Virtual Organization* specification must be converted into executable code for the agents platform.

It is also essential to mention E-Institutions[7] as a pioneering work in agent organizations, in this proposal the agents are structured in an organization (Electronic Institution), with regulatory mechanisms among its members, but the creating of new groups (dynamic organizations) is not allowed, ie, E-Institution mainly focus on the employment of organizational structures during the design process, and on the regulation of its agents, and it is very complicated to create new groups or sub-organizations dynamically at runtime.

MDD in MAS. Only some agent development methodologies have integrated the MDD techniques in the MAS design. The most relevant are: INGENIAS[8], TROPOS[15], PIM4AGENT[11] and AML[5]. All of this work is for the application of MDD to the agent modeling process, but the biggest problem found, is that they do not consider the development of *Virtual Organizations*.

An analysis of these approaches allows us to conclude that each methodology proposes its own models view with specific components which are in the majority of cases are different. Furthermore, some of the proposals do not achieve to the implementation phase, only defining high-level models, and difficulting enormously the developer work when tries to obtain executable code. Therefore, our purpose is to apply the MDD approach to *organizational-oriented* methodologies.

3 Modeling a Virtual Organization

This section presents some of the fundamental concepts of MDD and how this approach is used to model organizations.

MDD: Core Concepts. The MDD approach uses and creates different models at different abstraction levels, combining them when the application has to be implemented [14]. At high abstraction levels, the models are known as meta-models and they define the structure, semantics and constraints for a family of models (they are the model of a model). Models can be classified into three groups depending on their abstraction level: Computation Independent Model (CIM), which details the general concepts independently whether they are going to be implemented by a machine or not; Platform Independent Model (PIM), which represents the system functionalities without considering the final implementation platform; and the Platform Specific Model (PSM), obtained by combining the PIM with specific details about the selected platform.

A fundamental aspect of the MDD is the definition of sets of transformation rules between models, which allows the models to be automatically converted. Transformations are relational entities that describe how to map the rules concerning how the concepts of one model are transformed into the concepts of another model. These transformations can be applied at different abstraction levels. Horizontal transformations are applied over models that belong to the same level: PIM-to-PIM or PSM-to-PSM. Vertical transformations turn a general model into a more specific one (PIM-to-PSM) [14]. In general, all transformations are known as model-to-model transformations. Additionally, executable code can be automatically generated from a PSM. These transformations are known as model-to-code or model-to-text.

Virtual Organization meta-model. One fundamental challenge when defining a platform independent meta-model in an organization is selecting which concepts or components will be included in order to model the organization. It is almost too obvious to mention that this is not a trivial task, since it must define the minimum components necessary for the organization. To achieve this objective, some of the most well-known approaches in the area of MAS organizations were studied (mentioned in Section 2). The purpose of this analysis is to extract the common features from the methodologies studied and adapt them to the current proposal, specifying a platform independent meta-model of the organization. Therefore, the transformation mechanism turns the platform independent model (PIM) into platform specific models (PSM's). Figure 1 shows diagram that illustrates the relationship between the different models.

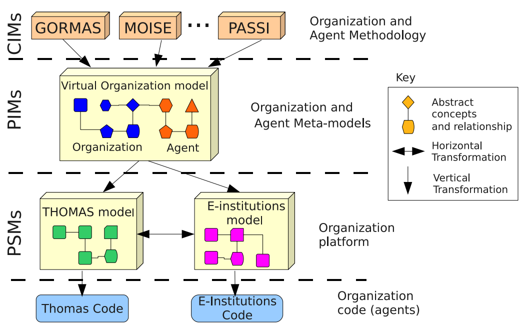


Fig. 1 Using MDD approach in Virtual Organizations

The main components and basic concepts employed in organization model are summarized in five meta-models: *Structural*, *Functional*, *Environment*, *Normative* and *Agent*. These meta-models are called πVOM (Platform Independent Virtual Organization Model). The relationships between the main components of this meta-model are described in detail in [2].

4 Using MDD for VO Design: Development Process

This section presents how to use the MDD approach to model *Virtual Organizations*. In order to do this, the development process for Virtual Organization using the MDD approach are explained.

Once the set of models that characterize the model of *Virtual Organization* has been completed (πVOM is described in detail in [2]), the process for transforming the VO into different platforms must be defined. The design process begins by selecting how abstract concepts (which are part of the unified organization model) are mapped to the target platforms. This paper focuses on the study of transformations on the platform that support agents organizations: THOMAS and E-Institutions. The transformation defines a set of mapping rules. The first mapping rules define which concepts of the source meta-model (πVOM) are transformed to which concepts of the target meta-model, a model-to-model transformation (PIM-to-PSM). The second transformation translates the models into the code templates of the organization (Application Launcher), which can be optionally combined with code written manually by the user. This is a model-to-text transformation (PSM-to-code).

In a more detailed explication the **development process** for a *Virtual Organization* using MDD is formed by a set of steps or phases (mainly transformations) that will obtain the executable code. In order to do these steps, a set of of tools, which support the process, are required. The steps employed at each design stage and their required tools are the following:

A. Model Creation: The developer creates diagrams (through graphical tools) which model the different units, roles, tasks, etc. of the developed system. To perform this step, the Eclipse IDE with a set of *plugins* is employed. These plug-ins are mainly *EMF*, *Ecore*, *GMF* and *GEF*, which allow the user to draw the models that represent the VO. Obviously, the needed meta-models (*Structural*, *Functional*, *Environment*, *Normative* and *Agent*)[2] has been previously loaded into the development environment in order to generate the appropriated VO models.

To illustrate this phase, a case scenario for making flight and hotel arrangements (a *TravelAgency*) is employed. The programmer must to draw (UML-Like) the VO that represent the *TravelAgency*. This scenario is modeled as an organization (*TravelAgency*) inside which there are two organizational units (HotelUnit and FlightUnit). Each unit is dedicated to hotels or flights, respectively. Two kind of roles can interact in the Travel Agency example: customer and provider roles. Figure 2 shows the *TravelAgency* structure, with its units, roles and relationships between them, using GORMAS notation. Similar diagrams must be created in this phase according to the different models that are part of πVOM.

B. Platform Selection: The developer must select which platform the user wants to execute different agents that form the system on the defined organization. In this step probably the agents or organizations can be executed in different platforms according to system modeling (scenario). For example, a possible scenario, is one where different ubiquitous agents running on various embedded platforms (PDAs or mobile phones) interact with *Virtual Organizations* to request different services.

Now, to do this, it is necessary to apply a model-to-model transformation (PIM-to-PSM). This is done using the Eclipse IDE and the ATL *plug-in* incorporating the appropriated set of transformation rules. It is important to remark that the same VO model can be transformed into different specific VO platforms. Table 1 illustrates the component transformations between two VO meta-models (from πVOM to

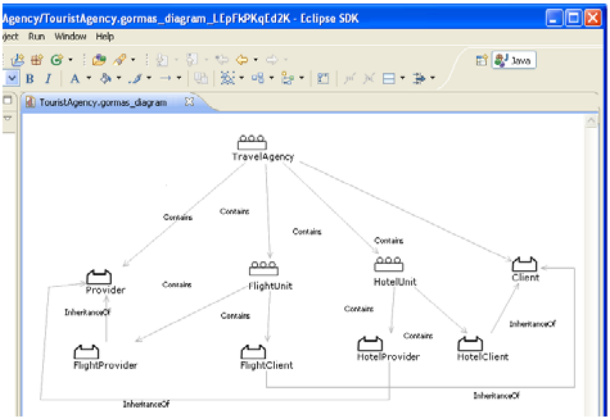


Fig. 2 Structural model of Travel Agency in πVOM

E-Institutions and THOMAS). These rules are a subset of the transformation rules needed in this phase, which are explained in detail in [1, 2]. In this way, VO concepts are mapped from source models to target models, and VO components are transferred, moved or changed from one model to another.

Table 1 Rules from πVOM to THOMAS and E-Institutions

πVOM Concept	Transformation Rules			
	Rule	THOMAS	Rule	E-INSTITUTIONS
Organizational Unit	1	THOMAS.OU	8	EI.Scene
Role	2	THOMAS.Role	9	EI.Role
Service	3	THOMAS.Service	10	EI.State $\in$ Scene
Norm	4	THOMAS.Norm	11	EI.Norm
RelationType	5	THOMAS.Process	12	EI.Transition OR EI.Illocutions
Resource	6	THOMAS.Resource	13	EI.World
Goal	7	THOMAS.Goal	14	EI.Norm

It is important to clarify that these rules are hidden from the developer (they are embedded in the application), and the programmer only uses when selecting the execution platform (PSM). The programmer automatically turns the VO model, designed in the previous step, a model that is based on E-Institutions. To illustrate how the rules are in the ATL language, the implementation of Rule 8 (only the first code lines) is shown in Figure 3.

This code generates all the *Scenes* (in E-Institutions) from the *OUs* (in πVOM), using the same Class Name. This code also shows the function `getAllRoles()` that examines all the *Roles* associated with each *OU*, which will be mapped to the

Roles Agent used in the different *Scene* (the function `getAllRoles()` will be used by the Rule 9).

```
helper context PIVOM!OrganizationalUnit
 def : getAllRoles() : OrderedSet(PIVOM!OrganizationalUnit) =
 self.children->iterate(child; hasRole: OrderedSet(PIVOM!OrganizationalUnit)=
    if child.oclIsTypeOf(PIVOM!Roles) then
          hasRole.append(child)
    endif
 );
rule OrganizationalUnit2Scene {
  from
       PIM : PIVOM!OrganizationalUnit(PIM.isOrganizationalUnit Root())
  to
       PSM : EInstitution!Scene (
          name <- PIM.name
        ...
```

Fig. 3 Organizational Unit rule in ATL language

C. Code Generation: The developer applies a model transformation to convert the designed models into code. To do this, it must use a PSM-to-code transformation. In this case *MOFScript* is employed which is an Eclipse plug-in that uses templates to do the translation process. These templates has been developed, as previously commented for E-Institution and THOMAS. From a practical viewpoint transformation/generation code, is to go through an XML file that describes the components and relationships of the meta-model and then generate another XML file that contains the specification of the E-Institution or THOMAS and which will be the application launcher.

In summary the Figure 4 shows that to design a VO use the following steps. First, we must create the model (UML-Like) of the organization (PIM). Second, we must transform the previous model to target model of the VO (PSM is obtained using a Model-to-Model transformation on ATL). Third, we must generate code for target platform (code is generated using a Model-to-Text transformation on MOFScript).

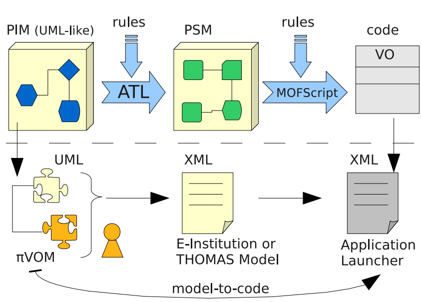


Fig. 4 Using MDD for VO design

5 Conclusions

This paper presents the application of the MDD approach for *Virtual Organizations* design. Through the application of MDD, the presented work can design and generate a first implementation of a Virtual Organization with agents that run over E-Institutions and THOMAS platforms (favoring inter-operability). Also, this work has checked the simplification of the VO design, since that some implementation details are hidden to the designer and that the process is automatic because it reduces the human interference in the *Virtual Organization* development. The automatic transformations allows generating code templates for different platforms using an unified VO model.

Acknowledgements. This work was partially supported by TIN2009-13839-C03-01 and PROMETEO/2008/051 projects of the Spanish government and CONSOLIDER-INGENIO 2010 under grant CSD2007-00022.

References

1. Agüero, J., Rebollo, M., Carrascosa, C., Julián, V.: Agent design using model driven development. In: 7th Int. Conf. on Prac. App. of Agents and MAS (PAAMS 2009), pp. 60–69 (2009)
2. Agüero, J., Rebollo, M., Carrascosa, C., Julián, V.: Developing virtual organizations using MDD. In: Proc. Workshop on Agreement Technologies (WAT 2009), pp. 130–141 (2009)
3. Argente Villaplana, E.: Gormas: Guías para el desarrollo de sistemas multiagente abiertos basados en organizaciones. Ph.D. thesis, Universidad Politécnica de Valencia (2008)
4. Carrascosa, C., Giret, A., Julian, V., Rebollo, M., Argente, E., Botti, V.: Service oriented multi-agent systems: An open architecture. In: AAMAS 2009, pp. 1–2 (2009)
5. Cervenka, R., Trencansky, I.: The Agent Modeling Language – AML. Whitestein Series in Software Agent Technologies and Autonomic Computing (2007), ISBN: 978-3-7643-8395-4
6. Dignum, V.: A model for organizational interaction: based on agents, founded in logic. Phd dissertation, Utrecht University (2003)
7. Esteva, M., Rodríguez-Aguilar, J.A., Sierra, C., Arcos, J.L.: On the formal specifications of electronic institutions. In: Sierra, C., Dignum, F.P.M. (eds.) AgentLink 2000. LNCS (LNAI), vol. 1991, pp. 126–147. Springer, Heidelberg (2001)
8. García-Magariño, I., Gómez-Sanz, J., Fuentes, R.: Ingenias development assisted with model transformation by-example: A practical case. In: PAAMS 2009, pp. 40–49 (2009)
9. Gateau, B., Boissier, O., Khadraoui, D., Dubois, E.: Moiseinst: An organizational model for specifying rights and duties of autonomous agents. In: EUMAS 2005, pp. 484–485 (2005)
10. Guessoum, Z., Jarraya, T.: Meta-models & model-driven architectures. In: Contribution to the AOSE TFG AgentLink3 meeting (2005)
11. Hahn, C., Madrigal-Mora, C., Fischer, K.: A platform-independent metamodel for multiagent systems. Autonomous Agents and Multi-Agent Systems 18(2), 239–266 (2008)
12. Jennings, N., Wooldridge, M.: Applications of intelligent agents. In: Agent Technology: Foundations, Applications, and Markets, pp. 3–28 (1998)

13. Jennings, N., Wooldridge, M.: Agent-oriented software engineering. In: Imam, I., Kodratoff, Y., El-Dessouki, A., Ali, M. (eds.) IEA/AIE 1999. LNCS (LNAI), vol. 1611, pp. 4–10. Springer, Heidelberg (1999)
14. Kleppe, A., Warmer, J.B., Bast, W.: MDA Explained: The Model Driven Architecture: Practice and Promise. Addison-Wesley Professional, Reading (2003)
15. Perini, A., Susi, A.: Automating model transformations in agent-oriented modelling. In: Müller, J.P., Zambonelli, F. (eds.) AOSE 2005. LNCS, vol. 3950, pp. 167–178. Springer, Heidelberg (2006)

Towards Soccer Simulation as a Testbed for Adaptive Systems and Agreement Technologies

Víctor Sánchez-Anguix, Juan A. García-Pardo, Ana García-Fornes, and Vicente Julián

Abstract. Agreement technologies and adaptive systems have arisen as promising mechanisms to design open dynamic multi-agent systems where there may be complex interactions and environmental conditions may frequently change. In this paper, we propose a development framework whose goal is to ease the deployment of adaptive techniques and agreement technologies in the soccer simulation domain. The framework provides a middleware that abstracts researchers from simulation details, thus they can focus on the particular study of their algorithms. Moreover, we analyze the soccer domain to identify how the types of adaptation and agreement technologies found in the literature are also present in the domain.

1 Introduction

In the last few years, the *intelligent software agent* paradigm has emerged as a proper technology for the deployment of complex systems: virtual organizations [2, 6, 14], new service oriented distributed systems [17], electronic commerce systems [7, 8], complex simulations [4, 9], and so forth.

Due to the fact that agents are situated in an environment, they must be flexible to every type of change that may be produced in the environment. Some works have focused in trying to provide agents with an adaptive behaviour to their environment [12, 13, 15, 18]. Even although agents may show adaptiveness at the individual level, they are part of a bigger scene: agent societies.

Agent societies are composed of agents. However, agents are not the whole part of the picture. As a matter of fact, agent societies have a social structure: roles, norms, global goals, interaction protocols, and so forth. Such agent societies are also situated in an environment, thus it is reasonable to think that their working may be also affected by changes produced in the environment. The inability to adapt to

Víctor Sánchez-Anguix · Juan A. García-Pardo · Ana García-Fornes · Vicente Julián
Universidad Politécnica de Valencia, Camí de Vera s/n, 46022, Valencia, Spain
e-mail: {sanguix,jgarciapardo,agarcia,vinglada}@dsic.upv.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 19–27.
springerlink.com

the environment may lead agent societies to an inadequate working of the system. Nevertheless, the adaptation task is far from being easy. Not only it must be decided when the adaptation must be carried out, but it is also necessary to determine which changes are to be produced in the society in order to adapt to the new environmental conditions. This usually implies that entities must coordinate about *how* and *when* to carry out these changes.

Agreement technologies [11] are a set of technologies whose objective is to success in the task of coordinating entities (i.e. agents) that are heterogeneous, autonomous, and therefore might present opposing preferences. Since agent societies can be formed by heterogeneous agents with opposing preferences, agreement technologies are proposed as the core conflict resolution mechanism for these situations. The benefit of the joint study of agreement technologies and adaptive agent societies is twofold. Firstly, it is interesting to study the existent relationships between agreement technologies and the possible changes that may occur in the agent society. For instance, it is interesting to study how agents evolve their negotiation strategies as the social structure changes. Secondly, agreement technologies can be the support technology needed to decide *when* and *how* to carry out the social adaptation.

In this paper, we propose a development framework whose goal is to ease the deployment of adaptive techniques and agreement technologies for open dynamic multi-agent systems in the soccer simulation domain. The framework provides a middleware that abstracts researchers from simulation details, thus they can focus on the particular study of their algorithms. The main advantages of the soccer simulation domain are: (1) it is easily understandable, thus it allows researchers to evaluate their work in a friendly and easy way; (2) The domain provides situations where the different types of adaptation described in the literature can be tested (norms, roles, goals, interaction patterns); (3) The main mechanisms that are part of agreement technologies can be evaluated in the soccer simulation domain (trust/reputation, norms, contracts, argumentation/negotiation).

The remainder of this paper is organized as follows. In Section 2, we briefly describe the soccer simulation framework and its abstract architecture. Then, in Section 3, we review how the framework provides a domain where the different types of adaptation and agreement technologies can be tested. Finally, we give some conclusions and lines of future work.

2 Abstract Framework for Soccer Simulation

Soccer simulation has been carried out with success by different researchers in the Robocup simulation league [8, 14]. The simulation focuses on a soccer match where a team composed of 11 players and a coach competes against a team formed exactly by the same number of players and coaches. The different players and coaches are agents that connect to a server which is the one to carry out the simulation of the environment according to the commands received from the agents. The server offers a perfect simulation tool for real-time multi-agent environments that are highly dynamic, have limited communication capabilities, and where uncertainty and imprecision problems arise as they would in a real physic system. This kind of simulation also allows to study coordination and cooperation techniques, decision-making, and learning [15, 16].

The Robocup soccer simulation domain has the sole goal of studying the dynamics of a soccer team during a match. However, it may be interesting to extend this simulation in order to take into account aspects of a soccer team that are not related to gameplay. The simulation may pass from a *gameplay level* to a *institutional level* where aspects of a soccer society, that may be interesting for adaptive techniques and agreement technologies, are simulated: matches, leagues, player markets, economic management, sports management, and so forth. Therefore, we propose a soccer simulation framework for the Magentix platform [1] that takes into account *gameplay level* and *institutional level* aspects. In the next sections, we describe our abstract architecture of the framework.

2.1 Gameplay Level

The gameplay level relates to every aspect of the environment that must be taken into account during a match. More specifically, we refer to the norms that regulate a soccer match and the simulation of a real soccer match (e.g. physics of the game). In our case, this game simulation is provided by the Robocup soccer server [10]. Furthermore, the DAInamite framework is used [5] in order to simplify the interaction with the soccer server, and to allow the abstraction of the robotic aspects of the players. The social aspects (e.g. norms, roles,...) which allow players and coaches to interact along the match are provided by the Magentix platform. An overview of the framework at the *gameplay level* can be observed in Figure 1.

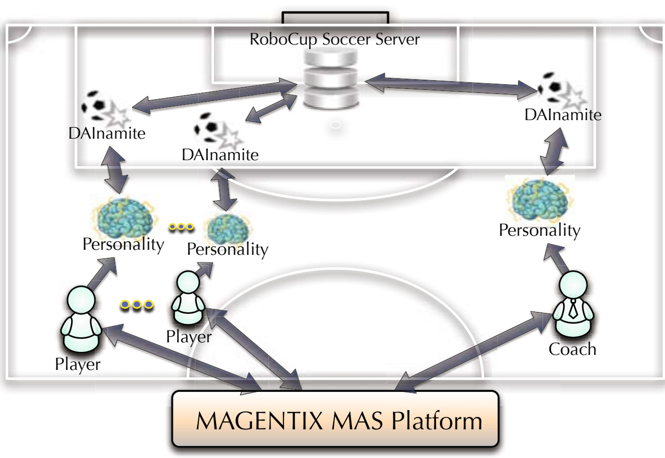


Fig. 1 Abstract architecture of the gameplay part

2.2 Institutional Level

The institutional level is related to every other aspect of the simulation which does not involve playing a soccer match. Every soccer team is situated in an environment where other organizations representing teams exist. Teams compete among them in different competitions in order to determine the winner of such competitions and/or achieve certain economic goals. Moreover, teams also interact with other teams outside the gameplay. For instance, they may agree economic contracts for a player transfer or they may form a coalition in order to negotiate their TV rights with the media. Additionally, there may be free players/coaches that may interact with teams in order to be contracted. Following, some of the aspects that are considered at the institutional level (see Figure 2) are described:

- Competitions: Soccer teams compete in different competitions that may have different rules (i.e. preconditions needed to compete, number of matches, how the winner is decided, etc.). Usually, teams earn money by winning matches in such competitions or ending the competition in a specific position.
- Fan base: They are a very important aspect to take into account by teams. In fact, some of the economic benefits of the team depends on the fans that decide to buy a match ticket. This aspect may be influenced by circumstances such as the sports results of the team, the joining of a new star player, and the importance of the match. The fan base is not represented by agents since its global behaviour is simulated by different rules/functions.
- Media: The media is also a very important source of economic benefits. They negotiate with the different teams in order to acquire their image rights. Of course, the result of the negotiation may be influenced by the popularity of the team. The media is not represented by agents, but its behaviour is simulated by the framework according to different rules/functions.

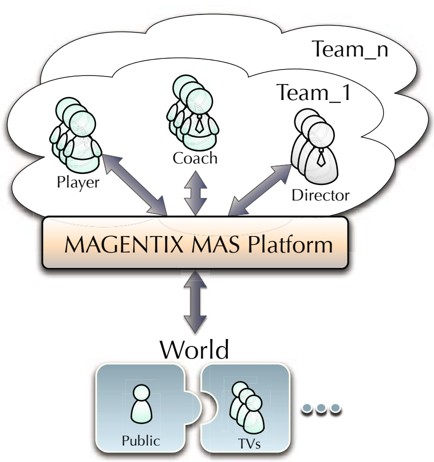


Fig. 2 Abstract architecture of the institutional part

- Free players/coaches: They are agents that are not contracted by any team. Therefore, they may be contracted by a team providing that both agree on contractual terms. Since it is an open system, these agents may appear at any moment.

Developers that use the framework are able to add new competitions, new teams, agents which that play some of the roles, and norms for the societies. Every aspect of the infrastructure can be be programatically modified. These modifications affect mainly to the institutional level, thus we discuss further in the next section the structure of a soccer team.

2.3 Basic Soccer Team

The key for the extended version of soccer simulation is the modelling of soccer teams as agent organizations. As a matter of fact, modelling a soccer team as an agent organization is not artificial at all, since real soccer teams are human organizations. However, when we talk about a soccer team we do not restrict ourselves to players and coaches, but we also include other roles such as directors. Therefore, the goal of the presented framework is to model an entire soccer team, both at the *gameplay level* and at the *institutional level*. Next, we give the basic structure of a soccer team offered by the framework. Due to space limitations, the team is just described by roles, norms and goals.

2.3.1 Roles

Basically, it is possible to identify three types of roles in a soccer team. These roles can be specialized according to more specific tasks, depending on the necessities of the developer.

- Players: The agents enacting this role are the only ones allowed to play the match. The DAInamite framework [5] is used as a middleware to interact with the Robocup soccer server. Each player may have a different personality. For instance, there may be agents that are selfish during the gameplay (i.e. they prefer to shoot instead of performing a pass), agents that desire fame, agents that have self-economic goals, etc. Players can also interact at the institutional level with different teams (i.e., negotiate its contract).
- Coaches: These agents are the ones with the goal of fulfilling the sports goals of the team. They devise strategies and formations that are to be carried out by players during the gameplay. The DAInamite framework [5] is used again as the middleware that interacts with the Robocup soccer server. They may also have different personalities during and outside the gameplay (i.e. some coaches are less risk averse towards the game, some coaches desire fame, some others want to fulfil some economic desires, etc.). Coaches can also interact with other agents and teams at the institutional level.
- Directors: They are the only role that does not interact with the Robocup soccer server. Therefore, their interactions are limited to the institutional level. They want to accomplish sports and economic goals of the team.

2.3.2 Team Norms

The inner norms of a soccer team regulate the behaviour of its agents. There are two types of norms that are related to the two different types of environments that can be found in the framework: *gameplay norms* and *institutional norms*. The first type of norms relates to the norms imposed by the coach role to the players during a match. Therefore, their validity is strictly restricted to a specific match (i.e., team formation). The second type are norms whose application only concerns aspects outside the gameplay. For instance, there may be economic rules that limit the salary of the players. It must be highlighted that team norms may frequently change along the lifetime of the team.

2.3.3 Team Global Goals

The goals can be classified into sports goals (i.e. win 20 matches, end at 3rd position at the end of the competition, win this match, draw this match, etc.) and economic goals (i.e. obtain 200k $ benefits, limit the cost of the players' salaries, etc.).

3 Agreement Technologies and Adaptation

In this section, we review how our proposed framework is useful to represent several situations in which agreement technologies and/or adaptive techniques may be applied to the system. First, we discuss how the different agreement technologies naturally arise in our framework. Second, several examples, where some social structural change is needed, are given. Different kinds of structural adaptation found in the literature are identified in these examples. Moreover, we discuss how these situations, where a social structure reorganization is to be carried out, are solved through agreement technologies. Additionally, agreement technologies may solve the problem of evaluating whether the team needs to reorganize or not.

3.1 Agreement Technologies

As it was stated above, agreement technologies are a set of technologies whose goal is to allow heterogeneous entities to cooperate in situations of opposing preferences. The main scientific topics related to agreement technologies are trust/reputation, norms, contracts, and argumentation/negotiation. Next, we describe where these technologies may be applied in our proposed framework:

- Trust/Reputation: On the one hand, a player may trust some teammates more that others based on past successful interactions. For instance, a player may prefer to pass the ball to a player who has been able to control the ball most of the times in past interactions. Similarly, a soccer team may trust more a player that has always fulfilled its contracts with the team than other players that try to be transferred to other teams whenever a good contract is offered. On the other hand, players and coaches may gain reputation by means of winning matches/competitions.

Therefore, these agents are better valued by the society and are preferred to other agents.

- Norms: Norms are used to regulate the behaviour of the agents that form the team. There are some obvious rules that all players must respect: gameplay rules and competition rules. However, there are some norms that the team may stablish in order to control the behaviour of the agents that are part of the team. For instance, the coach may stablish a norm that specifies that the formation for the current game is 4-3-3 and which players enact the different roles. Another example of norm application is the limitation of the players' salary so that the team is economically sustainable.
- Contracts: They are used in order to represent a set of obligations that must be respected by the signing parties. In our framework, contracts may represent player contracts, TV contracts, or even the social contract that the soccer team establishes with its followers (i.e. win the competition, do not descend of category, etc.).
- Argumentation/Negotiation: For instance, both techniques can be used to negotiate the transfer of a player between soccer teams, renegotiate contracts with current players, form a coalition with other teams in order to get a good economic agreement with the media, and so forth. At the gameplay, agents may be able to agree which textbook move should be employed in a corner kick, free kick, etc.

3.2 Adaptation

There are many social aspects which may change when agent organization need to adapt. Whether the organization is a *Designed MAS*, in which global goals are designed beforehand and the interaction structure is explicit, or it is an *Emergent MAS* organization, where the global behaviour is not designed and it emerges from the local interactions between agents, there are some organizational structural aspects which may require adaptation. In the inspiring work of Dignum et al. [3], some social aspects that may be adapted are identified: (1) Topology and roles, (2) society norms, (3) goals, and (4) interaction patterns. We present three examples in order to illustrate how the proposed framework is suitable to evaluate these structural adaptations:

• During the competition the team faces an economic crisis. Selling some of the players may help to overcome the crisis properly. The economic director, whose objective is to maximize money income of the team, may negotiate with the coach, who aims to keep the best players on the team. The adaptation may affect goals (new economic and sports goals), norms (limitations on players' salaries), and topology (the players that are part of the team).

• During a game the coach and his coaching staff decide to change the offensive strategy because they believe that the result is not favourable and there is not much remaining time until the end of the match. The adaptation may affect norms (formation), roles(more forward players), and interaction patterns (protocols that require less time).

• Finally, a team's star player does not play as well as it was expected. A decision needs to be made regarding the future of the player in the team. If the player is to be sold, the structure of the team may adapt to be able to fit to the changes (other player can enact its role, new players join the team, etc.). The directors from one side, and the coach from the other side, may negotiate to determine the outcome of the situation. This adaptation may affect the roles.

4 Conclusions and Future Work

In this paper we have presented a soccer simulation framework that aims to provide *gameplay* and *institutional* simulation of a soccer team. The framework is designed as a testbed for adaptive techniques and agreement technologies. In fact, the framework abstracts researchers from simulation aspects, thus it allows them to focus on their subject of study. The soccer simulation domain has proved to be suitable to evaluate the main types of agreement technologies and adaptation that are found in the literature. Also tracing and monitoring every action executed in the simulation helps either the developers to debug undesired behaviours and also the agents which can profit from this information to further learn how to adapt.

A prototype of the framework, with a limited functionality, has been implemented. As mentioned above, this prototype is based on the Magentix platform [1], DAInamite [5], Robocup soccer server [10] and the Java language.

Acknowledgements. This work is supported by TIN2008-04446, CSD2007-00022, PROMETEO/2008/051, TIN2009-13839-C03-01 funds of the Spanish government, and FPU grant AP2008-00600 awarded to V. Sánchez-Anguix.

References

1. Alberola, J.M., Such, J.M., Espinosa, A., Botti, V., García-Fornes, A.: Scalable and efficient multiagent platform closer to the operating system. In: Artificial Intelligence Research and Development, vol. 184, pp. 7–15. IOS Press, Amsterdam (2008)
2. Dignum, V., Dignum, F.: Towards an agent-based infrastructure to support virtual organisations. In: PRO-VE 2002, Deventer, The Netherlands, pp. 363–370. Kluwer, B.V., Dordrecht (2002)
3. Dignum, V., Dignum, F., Sonenberg, L.: Towards dynamic reorganization of agent societies. In: Proceedings of Workshop on Coordination in Emergent Agent Societies, pp. 22–27. Citeseer (2004)
4. Drogoul, A., Ferber, J.: Multi-agent simulation as a tool for modeling societies: Application to social differentiation in ant colonies. In: Castelfranchi, C., Werner, E. (eds.) MAAMAW 1992. LNCS, vol. 830, pp. 3–23. Springer, Heidelberg (1994)
5. Endert, H., Wetzker, R., Karbe, T., Heßler, A., Brossmann, F., Büttner, P.: The dainamite agent framework. DAI Labor, TU Berlin (November 2006)
6. Giret, A., Julian, V., Rebollo, M., Argente, E., Carrascosa, C., Botti, V.: An open architecture for service-oriented virtual organizations. In: PROMAS 2009, pp. 74–88 (2009)
7. Guttman, R.H., Moukas, A.G., Maes, P.: Agent-mediated electronic commerce: a survey. Knowl. Eng. Rev. 13(2), 147–159 (1998)

8. He, M., Jennings, N.R., Leung, H.-F.: On agent-mediated electronic commerce. IEEE Trans. on Knowl. and Data Eng. 15(4), 985–1003 (2003)
9. Helleboogh, A., Vizzari, G., Uhrmacher, A., Michel, F.: Modeling dynamic environments in multi-agent simulation. Auton. Agents Multi-Agent Syst. 14(1), 87–116 (2007)
10. Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I., Osawa, E.: Robocup: The robot world cup initiative. In: AGENTS 1997, pp. 340–347. ACM, New York (1997)
11. Luck, M., McBurney, P.: Computing as interaction: agent and agreement technologies. In: Proc. of the 2008 IEEE International Conference on Distributed Human-Machine Systems (2008)
12. Matos, N., Sierra, C.: Evolutionary computing and negotiating agents. In: Noriega, P., Sierra, C. (eds.) AMET 1998 and AMEC 1998. LNCS (LNAI), vol. 1571, pp. 126–150. Springer, Heidelberg (1999)
13. Narayanan, V., Jennings, N.R.: Learning to negotiate optimally in non-stationary environments. In: Klusch, M., Rovatsos, M., Payne, T.R. (eds.) CIA 2006. LNCS (LNAI), vol. 4149, pp. 288–300. Springer, Heidelberg (2006)
14. Norman, T.J., Preece, A.D., Chalmers, S., Jennings, N.R., Luck, M., Dang, V.D., Nguyen, T.D., Deora, V., Shao, J., Gray, W.A., Fiddian, N.J.: Agent-based formation of virtual organisations. Knowl.-Based Syst. 17(2-4), 103–111 (2004)
15. Stone, P., Sutton, R.S., Kuhlmann, G.: Reinforcement learning for RoboCup-soccer keepaway. Adapt. Behav. 13(3), 165–188 (2005)
16. Stone, P., Veloso, M.M.: Layered learning and flexible teamwork in robocup simulation agents. In: Veloso, M.M., Pagello, E., Kitano, H. (eds.) RoboCup 1999. LNCS (LNAI), vol. 1856, pp. 495–508. Springer, Heidelberg (2000)
17. Val, E.D., Criado, N., Rebollo, M., Argente, E., Julian, V.: Service-oriented framework for virtual organizations. In: ICAI 2009, vol. 1, pp. 108–114. CSREA Press (2009)
18. Weiß, G.: Adaptation and learning in multi-agent systems: Some remarks and a bibliography. In: IJCAI 1995: Proc. of the Workshop on Adaption and Learning in Multiagent Systems, London, UK, pp. 1–21. Springer, Heidelberg (1996)

Normative Argumentation

N. Criado, S. Heras, E. Argente, and V. Julián

Abstract. Virtual Organizations should evolve along time in order to face with unforeseen changing situations. This evolution implies the adaptation of norms, so agents must be able to consider dynamic norms which guide agent behaviours in their decision making processes. With this aim, this paper proposes the employment of argumentation protocols and schemes in order to allow agents to reach agreements inside dynamic organizations.

1 Introduction

Virtual Organizations (VOs) have been employed as an abstraction for modelling open agent societies. They include the integration of both the organizational and individual perspectives and also the dynamic adaptation of models to organizational and environmental changes [1]. In this way, it is necessary to detect situations of interest and to manage them maximising the adaptation flexibility and capacity. The organization adaptation implies, among other aspects, its norms, agreements, commitments and topologic structure.

Taking into account the challenges and difficulties arisen by the dynamic adaptation of VOs, our proposal consists on the adaptation of previous argumentation mechanisms and schemes in order to allow agents to reach agreements considering the existence of dynamic norms which control their behaviours. This paper is structured as follows: Section 2 provides a theoretical definition of VO together with a model of norms for controlling agents belonging to these VOs. Section 3 describes the proposed argumentation schemes for contract agreement. Finally, some conclusions and future works are shown.

N. Criado · S. Heras · E. Argente · V. Julián
Departamento de Sistemas Informáticos y Computación,
Universidad Politécnica de Valencia, Camino de Vera s/n. 46022 Valencia, España
e-mail: {ncriado,sheras,eargente,vinglada}@dsic.upv.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 29–36.
springerlink.com

2 Background

A *Virtual Organization* ($VO = \langle Ag, Act, Cap, Des, Prob, R, Roles, \leq_{prior}, N\rangle$) is a set of agents ($Ag$) that need to coordinate resources and services. Therefore, actions (Act) performed by agents are related to the execution and provision of services as well as resource accessing. This proposal does not specify a concrete agent architecture. However, each agent which participates in a VO is characterised with a set of capabilities, represented by the $Cap : Ag \rightarrow 2^{Act}$ function which maps each agents with the actions it is able to perform. In addition, there is one function that represents agents' desires (i.e. $Des : Act \rightarrow [0, 1]$ assigns a real value for representing desirability of actions) and another function that represents the expected probability of action occurrence (i.e. $Prob : Act \rightarrow [0, 1]$ assigns a probability to an action occurrence). Agents are related among them according to their roles (R). In this sense, $Roles : Ag \rightarrow 2^R$ is a function that matches an agent with the set of roles which it is currently playing. Priority among roles is expressed by a total order relationship ($R, \leq_{prior}$). Role rights and responsibilities are controlled by a set of norms (N).

Norms are a coordination mechanism that attempts to: promote behaviours that are satisfactory to the organization, i.e. actions that contribute to the achievement of global goals; and avoid harmful actions, i.e. actions that prompt the system in order to allow agents to be norm aware. In this work norms are classified into two main levels according to the norm scope: *institutional norms* (NI) and *interaction norms* (NC). The former type of norms has been explicitly promulgated by the institution; their violation is considered as an illicit act and enables sanctions. The interaction level is formed by both legal contracts and informal agreements between entities.

Definition 1 (Normative Context). The set of norms of a VO is defined as $N = NI \cup NC$.

Definition 2 (Institutional Norm). An institutional norm ($n_i \in NI$) is defined as $n_i = \langle D, C, T, S, E\rangle$ where:

- $D \in \{O, F\}$; being O and F deontic operators for representing obligations (O) and prohibitions (F), respectively.
- $C \in Act$ is an action that must be carried out in case of obligations; or that has to be avoided in case of prohibitions.
- $T \in R$ is the target of the norm; i.e. the role affected by the norm.
- $S \in Act$ is an expression which describes the action that will be carried out in case of norm violation. It is an enforcement mechanism employed for persuading agents to fulfil norms.
- $E \in Act$ represents the norm expiration action.

For simplicity, norms in our model have not an explicit activation condition. Thus, they are always active until the expiration action holds. Moreover, norms have been defined as a deontic control over a single action. However, this work can be extended for considering a more complex notion of norm.

Interaction Norms are normative prescriptions that emerge as a result of the interaction among agents. Therefore, this level of norms is formed by the contracts reached by agents along their execution:

Definition 3 (Contract). A contract ($n_c \in NC$) is defined as $n_c = \langle D, C, a, b, S, R, E \rangle$ where:

- $D \in \{O, F\}$ represents the deontic type of the norm.
- $C \in Act$ is the previously defined norm condition.
- $a, b \in Ag$ are the contract addressee and beneficiary agents, respectively.
- $S, R \in Act$ are expressions which describe the actions that will be carried out in case of contract violation or fulfilment, respectively.
- $E \in Act$ represents the norm expiration action.

Definition 4 (Institutional Conflict). Any contract n_c=$\langle D, C, a, b, S, R, E \rangle$ is in conflict ($IC(n_c, n'_i)$) with an institutional norm $n'_i = \langle D', C', T', S', E' \rangle$ if: i) they define opposite deontic constraints; i.e. $D <> D'$ and $D, D' \in \{F, O\}$; ii) over the same action $C = C'$; iii) the contract is addressed to an agent a which plays the norm target role ($T' \in Roles(a)$); and iv) the institutional norm is active ($\neg E'$).

Definition 5 (Contract Conflict). A contract $n_c = \langle D, C, a, b, S, R, E \rangle$ is in conflict ($CC(n_c, n'_c)$) with other contract $n'_c = \langle D', C', a', b', S', R', E' \rangle$ if: i) they define opposite deontic constraints; i.e. $D <> D'$ and $D, D' \in \{F, O\}$; ii) over the same action $C = C'$; iii) both contracts are addressed to the same agent $a = a'$; and iv) the previous contract is active ($\neg E'$).

3 Argumentation for Contract Agreement

Our main objective in this research is to solve a problem of contract agreement. In our initial settings we have two agents $i, j \in Ag$ that want to reach an agreement over the acceptance of a particular contract. In addition, to fulfil this contract, the *proponent* agent i has to persuade the *opponent* agent j to accept an interaction norm $n_c \in NC : n_c = \langle D, C, j, i, S, R, E \rangle$. Thus, the proponent agent i starts a dialogue with the opponent agent j and proposes it norm n_c. In this section, we show the type of arguments and attacks to these arguments that agents can interchange during the dialogue in the context of our normative model. Then, we propose a dialogue game protocol that controls the agents' interaction.

3.1 Initial Arguments

Following our normative model, the opponent agent j could have several possible types of arguments to reject the acceptance of norm n_c:

A1: There is an institutional norm $n'_i \in NI,\ n'_i = \langle D', C', T', S', E' \rangle$ which is in conflict with n_c (i.e. $IC(n_c, n'_i)$).

A2: There is a previous contract agreement by which agent j accepted an interaction norm $n'_c \in NC$, $n'_c = \langle D', C', j, k, S', R', E' \rangle$ which is in conflict with n_c (i.e. $CC(n_c, n'_c)$).
A3: There is a lack of interest in the reward awarded for the acceptance of this norm (i.e $Des(R) = 0$), but the opponent specifies which case of reward (R') it would be interested in so as to accept n_c.
A4: There is a lack of interest in the reward awarded for the acceptance of this norm (i.e. $Des(R) = 0$).
A5: Agent j cannot accept norm n'_c since it has no capabilities to fulfil this norm (i.e. $C \notin Cap(j)$).

Some of these arguments fit stereotyped patterns of human reasoning that argumentation theory has modelled as *Argumentation Schemes* [4]. Concretely, arguments A1 and A2 fit the normative version of the *Argument from Oppositions* scheme. If we adapt it to our settings, we have that:

Scheme 1 *"If the opponent agent j has accepted the norm n', with $n' \in \{n'_c, n'_i\}$ agent j cannot accept the conflicting norm n_c at the same time and at the same respect".*

Thus, due to argument A1 and A2, if agent j receives n_c and the same time has either an institutional norm n'_i or a previous interaction norm n'_c that conflicts with n_c, agent j concludes that it cannot accept the norm n_c.

If the opponent agent rejects norm n_c due to argument A3, it initially rejects the norm but providing the proponent with an alternative. This alternative would consist in a condition that, if fulfilled by the proponent, the opponent would accept norm n_c. In this case, the argument fits the structure of the *Two Person Practical Reasoning* scheme [4]. Again, adapting this scheme to our settings, it is reformulated as:

Scheme 2 *"Agent i intends to realize contract C with its associated norm n_c and tells agent j this. As j sees the situation, offering a new reward R' is a necessary condition for carrying out contract C, and j tells i this. Therefore, i should offer a new reward R', unless it has better reasons not to".*

With this argument the opponent points out the proponent a possible way of persuading it and carrying on with the dialogue. This scheme has a set of *Critical Questions* (CQ) [4]. They are stereotyped ways of attacking the conclusion of the argument and are analysed in the next section.

Finally, we assume that the preferences and capabilities of an agent do not change during the dialogue (but they can change when the dialogue finishes). Thus, A4 and A5 are indisputable arguments that end the dialogue.

3.2 Attacks

Depending on the type of argument that the opponent agent has for rejecting the acceptance of norm n_c, the proponent agent still has the opportunity of trying to persuade it to accept the norm. This is the case of arguments of types A1, A2 and A3. On the opposite, if the opponent does not provide the proponent with any alternative way of accepting norm n_c in A4 or it rejects the norm by putting forward A5, these arguments are considered indisputable and the dialogue will end here. Thus, arguments A1, A2 and A3 admit replies (i.e. attacks) from the proponent.

In the case of A1 and A2, the *Argument from Oppositions* [4] scheme does not have associated critical questions. However, it can still be attacked if the proponent agent thinks that there is an exception to its conclusion. This exception poses a new argument for not paying attention to the conflict between norms and the proponent could send it to the opponent, trying again to persuade it to accept norm n_c. This pattern of reasoning fits the *Argument For an Exceptional Case* [4]. Adapting this scheme, we have that:

Scheme 3 *"If the case of A1 (A2) is an exception, then the conclusion of the Argument From Oppositions scheme can be waived.*

In the context of our normative model, possible exceptions to A1 are:

E1: The desirability of the reward for fulfilling the norm n_c is greater than the desirability of the sanction applied for the non-fulfilment of the institutional norm n'_i (i.e. $Des(R) > Des(S')$).
E2: Actually, the sanction associated for the non-fulfilment of the conflicting institutional norm n'_i is never applied (i.e. *Prob(S) = 0*).

Also, possible exceptions to A2 are:

E3: The reward for fulfilling the norm n_c is greater that the sanction applied for the non-fulfilment of the interaction norm n'_c.
E4: The role r_i of the proponent agent i has more priority than the role r_k of the agent k that agreed the previous contract (n'_c) with the opponent agent (i.e. $\exists r_i \in Roles(i) : \forall r_k \in Roles(k) : r_i \geq_{prior} r_k$).

If the proponent agent attacks the argument A1 or A2 of the opponent agent by putting forward the conclusion of the *Argument for an Exceptional Case* [4], the opponent can surrender to the attack and accept the norm n_c or else can reject again the norm. A possible way of rejecting this attack could be to pose one of the critical questions of the scheme. If we adapt these questions to our normative model we have the following possible defeat[1]:

CQ1: Can evidence that the conclusion of the *Argument From Oppositions* scheme does not apply to this case be given?.

We have modelled possible attacks to A1 and A2 by adapting the type of attack defined in the CQ1. Therefore, the opponent can attack A1 or A2 by requesting the proponent to show evidence for the exception. Then, even if the proponent shows evidence for the exceptions, the opponent could still reject

[1] The original definition of the critical questions is available at [4].

the norm. Possible reasons could be, for instance, if it does not agree with (or believe) the evidence provided by the proponent, if it prefers observing the institutional norms, no matter how many times sanctions are applied or what the reward for fulfilling interaction norms is or if it has other argument of the type explained in the previous section to reject again the norm.

Finally, the proponent agent could attack argument A3. In this argument, the opponent is willing to accept norm n_c with the condition that the proponent should offer a new reward *R'*, following the conclusion of the *Two Person Practical Reasoning* scheme. We assume that offering a new reward *R'* is the only condition that the opponent puts for accepting contract C and its associated norm n_c. Thus, if the proponent agent decides not to offer a new reward, it just rejects A3 without giving alternatives for dispute. However, if it finally agrees to change the reward of the norm, it proposes again the new norm $n'_c \in NI : n'_c = \langle D, C, j, i, S, R', E \rangle$ to the opponent.

3.3 Dialogue Game Protocol for Contract Agreement

Following the approach of [3], we have modelled the agents' dialogue by means of a formal dialogue game for contract agreement that allows agents to reason about argumentation schemes. The protocol assumes the existence of a proponent agent i that tries to agree a contract C with an opponent agent j. As a result of this contract, j has to accept a new interaction norm $n_c \in NC$. The interaction protocol between the agents of the network has been modelled with the components for dialogue games identified by McBurney and Parsons in [2]. Figure 1 provides a graphical representation of the dialogue. In this figure, boxes represent steps and arrows represent transitions between steps, which are labelled with the associated locution.

R1: The proponent agent opens the contract agreement process proposing to the opponent the acceptance of the norm n_c ($propose(n_c)$).

R2: Opponent can *accept* the norm and end the dialogue, end the dialogue without providing an answer ($noCommit$) or reject the norm ($reject$).

R3: The proponent agent can either withdraw its proposal and end the dialogue ($noCommit$) or ask the opponent for providing an argument for the rejection of the norm ($why(reject)$?).

R4: The opponent agent can either withdraw the rejection and come back to R2 ($noCommit$) or *assert* an argument (i.e. A1, A2, A3 or A4) that supports the rejection of the norm.

R5: The proponent agent can either withdraw its proposal ($noCommit$) and end the dialogue or *pose* an attack with exceptions E1 or E2.

R6: The proponent agent can either withdraw its proposal ($noCommit$) and end the dialogue or it can ask the opponent for more information about its interaction norm n'_c that conflicts with the proposed norm. ($assert(norm)$?) By default, the proponent does not have any information about the sanction

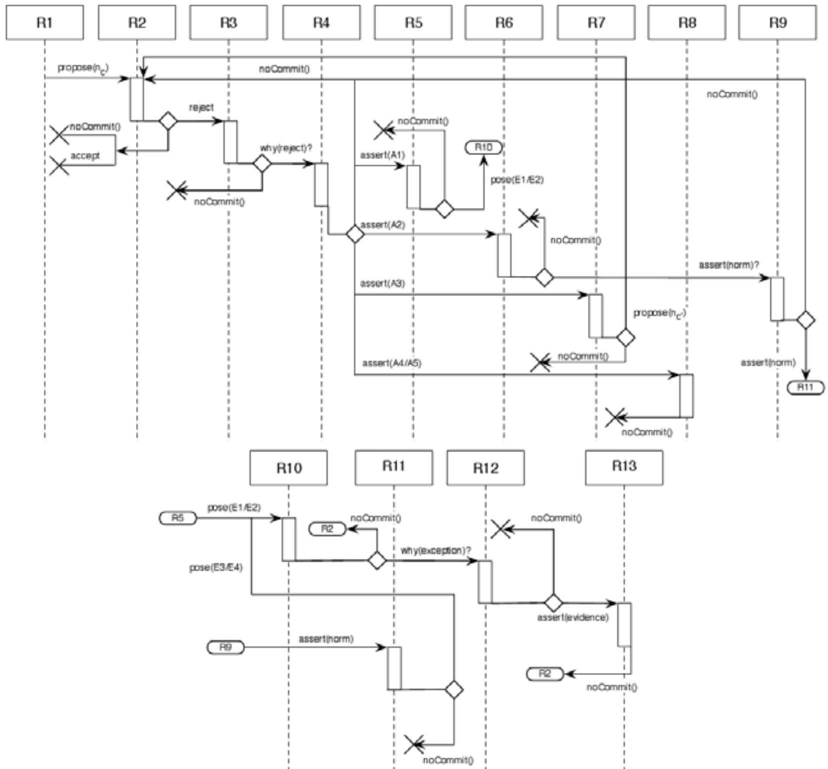


Fig. 1 Sequence Diagram of the Interaction Protocol

of this norm or the role of the agent that enacted it, which are necessary information to pose exceptions E3 or E4.

R7: The proponent can withdraw its proposal (*noCommit*) and end the dialogue or make a new proposal n'_c with the new reward R' ($propose(n'_c)$).

R8: The proponent agent has to withdraw its proposal and end the dialogue, since the opponent does not give any chance for dispute.

R9: The opponent agent can either withdraw the rejection and come back to R2 or provide the proponent with information about its previous interaction norm that conflicts with the proposed norm ($assert(norm)$).

R10: The opponent agent can either withdraw the rejection and come back to R2 or ask the proponent for providing an argument for the exception attack received ($why(exception)?$).

R11: The proponent agent can either withdraw its proposal and end the dialogue or *pose* an attack with exceptions E3 or E4, coming back to R10.

R12: The proponent can either withdraw its proposal and end the dialogue or provide the opponent with an evidence that supports the exception attack that it posed ($assert(evidences)$).

R13: The opponent agent withdraws its rejection and comes back to R2.

4 Conclusions and Future Work

VOs should evolve along time in order to face with unforeseen situations. Thus, agents must be able to reach agreements taking into account the existence of dynamic norms which control their behaviours. This work proposes a new dialogue game protocol together with argumentation schemes for contract agreement inside dynamic VOs controlled by norms. As future work, we plan to evaluate this approach through different case studies.

Acknowledgements. This work was partially supported by the Spanish government under grants CONSOLIDER-INGENIO 2010 CSD2007-00022 and TIN2009-13839-C03-01, by GVA funds under PROMETEO 2008/051 project and by the FPU grant AP-2007-01256 awarded to N. Criado.

References

1. Dignum, V., Dignum, F.: A landscape of agent systems for the real world. Technical report, Inst. Information and Computing Sciences, Utrecht University (2006)
2. Mcburney, P., Parsons, S.: Dialogue games in multi-agent systems. Informal Logic 22, 2002 (2002)
3. Reed, C., Walton, D.: Argumentation schemes in dialogue. In: Dissensus and the Search for Common Ground, pp. 1–11 (2007)
4. Walton, D.: Appeal to Expert Opinion. Penn State Press (1997)

Engineering Ambient Intelligence Services by Means of MABS*

Teresa Garcia-Valverde, Alberto Garcia-Sola, Francisco Lopez-Marmol, and Juan A. Botia

Abstract. In this work, the methodology AmISim to test and to deployment of Ambient Intelligence (AmI) system is presented. The development of AmI systems is a complex task because this technology must adapt to users and contextual information as well as unpredictable and changeable behaviours. So, we focused in how the methodology AmISim can help to the engineering of adaptative services for users. In this case, we propose a predictor of location based on Hidden Markov Models (HMMs). So, the system can offer Location-Based Services(LBS) that adapt to the users. To this end, we propose a methodology based on a previous social multi-agent based simulation (MABS) and a following deployment of the service in a real environment.

1 Introduction

Ambient Intelligence (AmI) is a new vision in which people are surrounded by intelligent embedded objects within an environment that is able to recognize and to respond to different individuals [2]. The contextual information and ubiquitous computing are the base of AmI. From ubiquitous computing, AmI environments have many non-intrusive and invisible devices communicated and integrated into the environment. These devices generate information about the environment, users and changes in both of them; generating a context-aware environment. With this contextual information, services and applications in AmI systems are able to adapt to changes.

University of Murcia
e-mail: {mtgarcia,agarciasola,pacolopez,juanbot}@um.es

* This research work is supported by the Spanish Ministry of Science and Innovation under the grant AP2007-04269 of the FPU program, the Séneca Grants program through the 09493/FPI/08 grant and in the scope of the Research Projects TSI-020302-2009-43, TIN2008-06441-C02-02, through the Fundación Séneca within the Program "Generación del Conocimiento Científico de Excelencia" (04552/GERM/06).

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 37–44.
springerlink.com

The development of AmI environments is a difficult process because of the high variety of scenarios, heterogeneous devices, different sources of information, etc. So, the design and development of a framework that allows testing, validating and deployment AmI environments is needed. Normally, this implies using living labs [3] (imitations of the real environment). However, this supposes a considerable economical investment and it is unfeasible in many situations (e.g., an scenario of evacuation emergency). Because of this, several simulators in related fields to AmI have been developed. However, these simulators are focused in just a part or few parts of an AmI environment, but they do not offer a entire framework to AmI simulation.

In order to approach these problems, we propose a hybrid approach between real and simulated environments to test, validate and deploy AmI systems. So, we use AmISim[4], a simulator of entire AmI environment which uses Ubik [7]. Ubik is the simulated part of AmISim, a MABS which has capabilities to simulate the physical world and a high number of agents, their interactions and individual or group behaviours of agents. The Ubik simulator is integrated with the real part of AmISim, a entire framework which is able to use semantic technologies for modelling the context model and uses it to provide adaptive and intelligent services to users. Finally, we can deploy over a real environment all real parts tested and validated with AmISim. The use of the hybrid methodology of AmISim is illustrated through a practical scenario in which a service can predict locations of workers in a office building. This prediction is based on Hidden Markov Models (HMMs). So, the system can provide nearest features of interest to users and adapt to their requirements using Location-Based Services(LBS) and other contextual information about them.

The next section describes the architecture of AmISim. Section 3 describes the methodology used to develop AmI services. We present the simulation scenario used in this work, the followed methodology for model learning and service building in AmI environment from a MABS and an approach based on a machine learning technique, HMMs, for predict locations. Finally, in section 4 some conclusions are given.

2 AmISim Architecture

In this section the generic model for simulation of AmI systems, AmISim, is described. The architecture of AmISim has a layered structure. Typically, an AmI system follows also such layered disposition. In the bottom, we find a layer corresponding to the physical environment in which the artificial system is subsumed (e.g. a physical building). We include there the users (e.g. the workers at the building). On the second layer, we find the hardware appliances used to both sense from the environment (i.e. sensors) and act on it (i.e. actuators). At the third layer, the context-awareness layer is found. It is in charge of maintaining an up-to-date context of the users by using static information about the environment (e.g. the structure of the building) and dynamic one (e.g. location information of users). The top layer AmI includes services and applications, which use information from the third layer to provide adaptive and intelligent behaviour to users.

We have used Ubik to social simulation models to reproduce both the first and second layers of the architecture. We simulate physical spaces, sensors, actuators and agents working or living in such spaces. Ubik has been developed with MASON[1] and it is precisely the simulated part to test and validate our AmI system. On the top of that, we collocate OCP(Open Context Platform)[5], a real context-aware management system which gathers data coming from simulated sensors and send commands to simulated actuators. The OCP middleware stores the information in an OWL-based ontology model and interprets it. Again, on the top of OCP, we build our specific services and applications, which use contextual information from OCP.

Services in the adaptation model are divided in basic and advanced. Basic services are designed and built to assist complex services in providing adaptive behaviour. Examples of basis services are location, preprocessing contextual information and so on. In the other hand, techniques to build complex services are using neural networks or SWRL(Semantic Web Rule Language) rules to model user behaviour, applying mining data streams techniques, etc. Advanced services and applications act over the user and the environment. Basic services are used as building block for advanced ones.

3 An Example of AmI Service Development

Location information is one of the most essential and used contextual information in context-aware systems [8]. With information about location, system can provide nearest features of interest to users and adapt to their requirements, i.e., Location-Based Services(LBS). Users usually have some degree of regularity in their motion. So, the system can predict their motion and acts proactively.

There are plenty of technologies able to automatically identify and inform about the location of people and objects. One of the most spreaded is RFID. A RFID system consists of tags, which are, usually, passive wireless elements, attached to persons or objects. Then, fixed wireless readers collocated within the environment, retrieve information from those tags, identifying people and objects as they are perceptible by them.

RFID based location systems work in a simple way: when a person or an object pass near a RFID antenna, person or object location is associated with that of the antenna. By using machine learning, we may improve such system in two ways. In the one hand, we may experiment with the fixed location of the antennas, in order to obtain the best location at the same time we try to optimise their number. In the other hand, it could be possible,

[1] MASON, http://cs.gmu.edu/eclab/projects/mason/

probabilistically, to augment the resolution of the location system (i.e. going further than giving only antenna locations).

3.1 Methodology for Model Learning and Service Building

In this work, a hybrid methodology to model learning and service building is proposed. The use of social simulation to build adaptative and advanced services in AmI environments allows us to obtain a set of simulated data according to the real environment. So, we can built different algorithms from this set of data and validate our models. When the algorithms and models are checked, we can deploy the services over a real environment.

In the first phase of the methodology we compound a virtual representation of the real environment in which the AmI system will be deployed. Ubik[2] is used for this. Ubik models the physical building, sensors and actuators, which are collocated in specific areas, tight to reality when the real application is installed and deployed.

In the second phase, we model people as users by using agents. The kind of services dictates the aspects of users that we will modeled (e.g. if we develope a fire evacuation service, users should be able to run and leave the building). In this paper, we are interested on building an indoor location system. Thus, users should evolve freely in their working time through the building in a realistic way. The third phase is in charge of developing algorithms providing such behaviour using the virtual environment generated in the first two phases. In the fourth phase, such algorithms are tested by using off-line data and directly by using the simulator with the algorithm implemented as a service.

The following sections describe the different phases to built and test an AmI service according to this methodology. This service can predict the location of the agents in an indoor environment. Finally we can successfully deploy the proposed service in a real environment.

3.2 Environment and Agent Models

As we described above, our location approach is based on indoor environments. So, we need simulate the environment and agent models. We use Ubik to simulate these models. On the one hand, in the environment model, we propose an office building scenario which consists of a configurable number of floors, stairs, rooms, corridors, offices, a room to relax with a coffee maker, a bathroom and a meeting room. In each office, several agents workers have their desks with their own fixed phones. Each agent wears a RFID tag which identify him and the building is equipped with RFID antennas in some places

[2] http://ants.dif.um.es/staff/emilioserra/ubiksim/

where it is interesting to locate people. When an agent pass through an antenna, a message is generated reporting his current position to the context middleware OCP. On the other hand, the agent model is a fundamental goal in this kind of simulator (focussed on users). We propose MABS for achieving this goal. Each user is simulated with an agent and we can simulate complex multi-agent model that simulates physical and human behaviour related features of agents, their interactions and individual or group behaviours.

We probabilistically model the behaviour of users. In our proposal, behaviours are defined as situations the agent should play in each moment. Transitions between behaviours are probabilistic. The underlying model is a hierarchical automata (i.e. in a higher level there is a number of complex behaviours that the agent may play and once it is in a concrete state, within the state there is another automata with more simple behaviours). So, the modeling of each behaviour is treated separately and the modeler is abstracted of unnecessary details. So, in the lowest level (basic actions), each state is atomic. In the figure 1 a graphical example of the automata is showed.

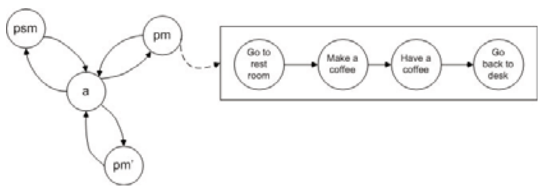


Fig. 1 Hierarchical Automata

We can suppose, for a working day, a **basic state**, a, *go to work*. The agent remains in the basic state most of the time. As we showed in the figure, the action of the basic state can be interrupted in order to go other states, for example, *answering a phone call* (psm), *having a coffee* (pm) or *going to the bathroom* (pm'). These states are of two types: psm, **probabilistic without memory** and pm y pm', **probabilistic with memory**.

In the pm states, the agent can go from the basic state to this state (and go back) as many times as the call events occur, regardless of how many times has visited the state. So, the probability of going from the basic state S_{sb} to this kind of states can be modelled by a exponential distribution of the parameter λ, i.e., the probability density function at time t is $f(t) = \lambda e^{-\lambda t}$. However, the pm' states are the states that depend of the times that the state had occur in the past. For example, go to the state *having a coffee* is more probable if the agent had not had coffee yet. So, there is a delay between each activation of the function that modelled the behaviour. In fact, we use a conditional probability density function, derived of the exponential. Let t_0 be the time until that it is not possible to generate an event, then $P[T >= t|T > t_0] = 1 - e^{-\lambda(t-t_0)}, t > t_0$. Finally, the spent time in a particular state is modelled with an exponential where the parameter λ describes the mean spent time.

Some mentioned behaviours imply movement through the office building. To solve the movement the agents, they have between their beliefs an abstract schema of the map of the building. The abstract schema of the map of the building defines the structure of the building with areas enough big, as room, corridor, etc., but not in a low level like a cell of a grid because this will be very inefficient. The abstract schema of the map of the building is built by an undirected graph with weights, which models the structure of the building. So the topology of the building is based on a graph $G = (V, E)$, where V is the set of rooms, corridors and stairs, E is the set of doors and crossings and $v, v' \in V$ are connected by the edge $e \in E$ if they are connected spaces in the building. Each edge has a weight X that defines the distance between the nodes that connects.

This abstract schema is stored in the domain ontology used by OCP (static context about the environment). So, agents can access to the structure of the building and to use it for calculating the best path for their movements using the Dijkstra algorithm [1]. When the agent knows the best path to a particular destination, this path is added to its beliefs, like in a real scenario.

3.3 Approaches for Indoor Location

In this section we face the problem of locating the RFID(Radio Frequency Identification) users in indoor environments. As we mentioned above, it can not be assured at any precise moment t the location of an agent. In these lack of information situations, the only available data are the generated data by the antennas when the workers go past near them. Data will be compound by traces. Traces will be a list of antenna locations the user pass through until he reaches its final destination. Each time the user leaves her desk, she is supposed to go somewhere. The list of antennas she traverses, generates a trace. We may represent the set of traces of a simulation with $\langle(t_{1,0}, a_{1,0}), (t_{1,1}, a_{1,1}), \ldots, (t_{1,n}, a_{1,n}, l_1)\rangle$, $\langle(t_{2,0}, a_{2,0}), (t_{2,1}, a_{2,1}), \ldots, (t_{2,m}, a_{1,m-1}, l_2)\rangle$, $\ldots\}$, where $(t_{i,j}, a_{i,j})$ indicates that a user passed near $a_{i,j}$ antenna, at time $t_{i,j}$. Besides, l_k is a concrete final location (e.g. the bathroom or a meeting room) if $t_{k,r} < \tau$, where τ is the threshold for a given location and $t_{k,r}$ the time that the agent spends in a specific location. Such traces may be used (see next section) to perform machine learning and predict user location more effectively.

3.4 An Approach Based on HMM for Location Service

To location prediction we propose the use of a probabilistic framework based on HMM. An HMM is a doubly stochastic process with an underlying stochastic process that is not observable (it is hidden)[6]. The basic idea is to define a set of hidden states, where each state has an associate probability distribution over a set of observables states. In each hidden state, observable information is showed according to these probability distributions. The hidden states are interconnected by probabilistic transitions between them. HMMs have an high tolerance according to the narrowing or lengthening of the time sequences. So, we can abstract to the model of specific features of agents like the speed of their motions.

In this prediction location model, $S = \{S_0, S_1, \ldots, S_N\}$ is the set of hidden states, where S_t is the hidden state at time t. These hidden states model the different possible locations according to the nodes of the graph stored in the domain ontology. In this model there is a final state, the destination of the agents that is obtained by the traces of the antennas (see section 3.3).

Probabilistic transitions between hidden states are modelled by using the edges of the graph in the ontology domain representing the building. Transition probabilities between hidden states, $A = a_{ij}$ for the probability of going from the i to the j location, is defined as the transition probability between the hidden state s_i and s_j if in the graph exits a edge that interconnected these hidden states, and $a_{ij} = 0$ in other case. Finally, $O = o_0, o_1, \ldots, o_m$ is the set of observables states, with o_t the observable state at time t. The observables states O are the sequence of active RFID antennas in the trajectory of an agent.

In this framework, firstly the traces of antennas are analyzed for detecting the most frequent destinations of users. So, we classify groups of traces for learning models according to the most frequent destination. A HMM is built for each destination, where the HMM models the possible paths from the desks of agents to their destination.

Finally, each model is learned by using their group of training sequences (i.e. traces with the same destination). We estimate the HMM parameters by using the Baum-Welch algorithm. Thus, learning the HMMs consists on finding the maximum-likelihood estimate of the parameters of the HMM given a set of observed feature vectors (group of training samples for a single destination). When the HMMs are estimated with the offline training sequences, we can test them in AmISim over simulated motions of agents. To predict the future location of the agents we use the forward-backward algorithm. This algorithm computes the probability of a specific observation sequence. So, we can estimate the probability of a particular trace of active antennas for an agent in all the HMMs of the system. Given a sequence, the HMM with the higher probability for such sequence allow us to locate the user.

4 Conclusions

We propose a methodology based on the use of multi-agent based simulation for model learning and service building in AmI environment. The methodology proposed follows a process of refinement which allows us to built algorithms over a offline data set from the multi-agent social simulation and to validate and to fit these algorithms over a real environment. We design a MABS in which agents are modelled with probabilistic behaviours in a office building. So, they do usual routines in a day of work and they learn paths and add them to their beliefs as a real user would do. This simulation allows us to use a machine learning technique based on HMMs to predict the destinations of the agents. So, we can also predict accurately the movements of real users within an office building and built AmI advanced services for adapting to them.

In future works will design an Hierarchical Hidden Markov Model. So, we can model the transitions between the HMM for each specific destination. Then, the system can predict the frequency of change between destinations (and behaviours). This model covers aspects as that if an agent goes to the bathroom, it is not probably that the agent repeats this path during some time.

References

1. Dijkstra, E.: A note on two problems in connection with graphs. Numerische Mathematik 1, 269–271 (1959)
2. Ducatel, K., Bogdanowicz, M., Scapolo, F., Leijten, J., Burgelman, J.C.: Scenarios for ambient intelligence in 2010. Technical report, ISTAG, The Information Society Technology Advisory Group (2001)
3. Litz, L., Gabel, O., Reif, M.: Ncs testbed for ambient intelligence. In: IEEE International Conference on Systems, Man and Cybernetics, vol. 1(115), p. 120 (2005)
4. Garcia-Valverde, T., Serrano, E., Botia, J.A., Gomez-Skarmeta, A., Cadenas, J.M.: Social simulation to simulate societies of users inmersed in an ambient intelligence environment. In: SS@IJCAI 2009 (2009)
5. Nieto, I., Botia, J.A., Gomez-Skarmeta, A.F.: Information and hybrid architecture model of the ocp contextual information management system. Journal of Universal Computer Science 12(3), 357–366 (2006)
6. Rabiner, L.R.: A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE 77, 257–286 (1989)
7. Serrano, E., Botia, J.A., Cadenas, J.M.: Ubik: a multi-agent based simulator for ubiquitous computing applications. Journal of Physical Agents 3(2), 39 (2009)
8. Weiser, M.: Ubiquitous computing. IEEE Computer 26(10), 71–72 (1993)

Mobile Agent Middleware for Intelligent Management of Communication Infrastructures

Carmelo R. García, Francisco Alayón, and Ricardo Pérez

Abstract. A case of middleware for agents that work in mobility scenario is explained in this paper. Using this middleware the agents achieve an intelligent management of the communication infrastructure. The principles of operation of the middleware are inspired in the ubiquitous computing paradigm. Specifically, a design of a middleware is described in this work, permitting a proper integration of mobile agents in the corporative information system. In this context, the concept of proper integration means that all the information that is related with the mobile agents operations is available at the moment, and in the required amount by all the processes of the information system of the organization.

Keywords: middleware, mobile computing, ubiquitous computing.

1 Introduction

Nowadays, systems that use mobile agents are frequent in fields such as e-commerce, e-government, network administration, etc. In general, mobile agents play an important role in distributed systems and networks. A mobile agent is an element which works in the information system on an autonomous way, using the resources provided by the system, especially those that are required to work in an integrated manner in spite of its mobility. The advances in the information and communications technologies have allowed to automate processes, which are executed in mobile scenarios, in the information systems of the organizations and to put in practice models that integrate the different stages of the operations of the information system.

The middleware described in this paper is suitable for corporations where productive and control activities are performed in geographically dispersed areas using mobile agents, permitting this middleware the proper integration of the mobile agents in the corporative information system. Broadly, we are talking about

Carmelo R. García · Francisco Alayón · Ricardo Pérez
Cybernetic Science and Technology
University of Las Palmas de Gran Canaria
Campus Universitario de Tafira, 35017, Spain
e-mail: `{rgarcia,falayon,rperez}@dis.ulpgc.es`

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 45–52.
springerlink.com

scenarios such as logistic, transport, etc., where the accomplishment of its productive and control activities require data that give information about where, when and how this productive activity is carried out, being these activities performed by mobile agents with the capacity of autonomous and intelligent interaction with different communications infrastructure and with different contexts.

2 Related Works

Assuming the ubiquitous system as a set of mobile and autonomous agents that collaborate to transform a physical space in order to provide a shared data useful for individual users or organizations, related systems can be found revising the bibliography. Considering as a criteria the dimension of the physical space where the agents work and the number of agents in the system, the ubiquitous computing environment can be classified in: smart environments, for example the Gaia system [1] is a case of this smart environment oriented to rooms, houses or buildings, or corporative environment, AnySpot [2] is a platform for pervasive access and sharing of documents, or finally, urban environments, such as Navitime system [3]. In all the types of systems mentioned above, there are two key aspects that are: the first, the context aware computing, and the second, how the mobile and autonomous agent can be integrated in the systems to achieve a collaborative computing. Different approaches have been suggested for context-awareness between agents, these approaches can be grouped in two categories. The first consist of a framework for the development and execution of sensor-based context aware agents, providing a set of reusable components [4]. The second consist of developing an infrastructure for context-awareness named middleware and more specifically a middleware for context-awareness. Additionally, this second approach has the advantage to fulfill the second critical aspect mentioned above. Examples of cases of middleware for context-aware agents in ubiquitous environments are the proposed by Ranganathan and Cambell [5], LIME [6], XMIDDLE [7] and RCSMl [8]. The middleware explained in this paper is oriented to corporative or urban environment and is a case of middleware for context awareness, additionally a relevant characteristic is the intelligent managements of the communications infrastructure in order to optimize the economic costs of use of these infrastructures because private and public communications infrastructures are managed by itcentralization of services.

3 General Description of the Operation Context

According the ubiquitous computing paradigm, the mobile agents must be able to execute on autonomous way in changing contexts, adapting to the persons in order to achieve a natural interaction between them. Using this middleware, the mobile agents, executing on different platforms of the corporation infrastructure, perform important tasks, producing quality data. Quality data means data obtained and processed in real time and in the proper amount. The figure 1 shows a general vision of the system from the point of view of the mobile agent devices.

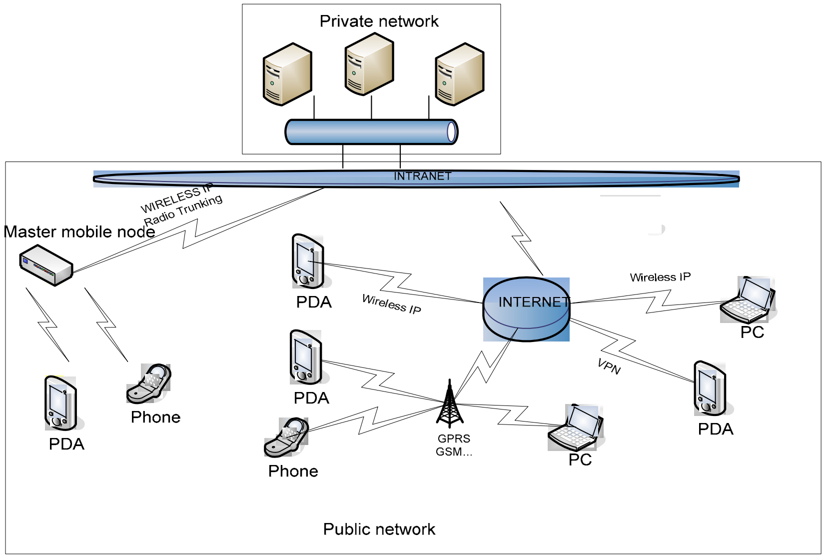


Fig. 1 General vision of the system

In the system, figure 1, PDAs, cellular phones, contactless card, mobile personal computer, etc. are the platforms where the agents are running. For example, clients can use the contactless cards or the cellular phones to pay a service provided by the company, or the maintenance staff uses PDAs or personal mobile computers to interact with others devices. In the system architecture, an element named master mobile node has the more important role between the ubiquitous devices, because it plays a main role with respect to other system elements. It can act as repository where the agents store data or from where they can load new distributions of data and programs. To do this data transfer, the agents use the services provides by the middleware run-time. A second relevant role of the master node consists of the network integration of the others ubiquitous devices. Additionally, and as a consequence of this integration, the master nodes can control to other devices, allowing an intelligent management of the rest of elements o the system infrastructures (hardware, software and agents).

In a ubiquitous context, the network is responsible of the data transfers between agents which are running in mobile and fixed platforms. In our model of system, the ubiquitous network is made up by different communication infrastructures: the long distance mobile communications can be accomplished for example by radio, GSM or UTMS infrastructure, the local mobile communication is performed by IEEE 802.11 wireless networks operating in a set of sites located in the geographic scope covered by the company and very short distance is achieved by Bluetooth and RFID networks operating places or mobile platforms of the company. Therefore, the ubiquitous network integrates a mixed infrastructure managed by the middleware; that decides what type of infrastructure to use considering different criteria such as: infrastructures availability, costs, relevant of the data to be transferred, etc.

The figure 2 shows an abstraction of the general functional structure of these kinds of corporative systems. In the top level, the intelligent agent make use of the mobile computing services, these agents are running in mobile platforms located in a dispersed geographic area where the corporation works. These mobile intelligent agents have all the resources (hardware and software) required to work in autonomous way; as a consequence of this capacity of autonomous execution the permanent connection, using communication infrastructure, with the central services is not required. The agents not only perform all the tasks related with the activity of the corporation (for example production, fare collection, control planning and customer information), but also they support the control of the technical status of the hardware and software resources of the system, including the intelligent management of the communication infrastructure, and if an exception occurs then the communication with the control center of the company is made. In this context, an exception is any event that can affect to the operations of the agents or to the correct work of the resources (hardware and software). The role of the middleware is to provide to the agents with transparent communication channels regardless of the communication infrastructure used, for example, long distance mobile communication infrastructure versus local mobile communication infrastructure. Another role of the middleware is to enable the spontaneous interaction between agents using the available communication infrastructures. The localization of the agents is very important for the middleware, because this information can be used to choose the suitable communication infrastructure in order to fulfill requirements related with the operational cost of the system, for example, if the positioning of the agent is inside of a area covered by the local net-work infrastructure, the data transfers can be accomplished using this infra-structure because the use of it is free of charge from the economic point of view.

In a general scenario, corporations that require to carry out activities in mobile environments by autonomous agents, the agents support all the tasks related with the activity of the corporation, so these agents can be classified as:

- Production agents, these perform all the tasks required to provide the services offered to the clients.
- Planning control agents, these perform the supervision and control of the operations in order to guarantee the fulfillment of the planning.
- Client information agents, they are responsible to inform to the clients about the services of the company, for example timetables, incidents of the services, payment systems, etc.

In order to guarantee a secure and proper work of the agent:

- Maintenance agents, they supervise the hardware and software resources in order to detect technical failures and to update, in an unsupervised and automatic way, the data and programs of the different agents.
- Transactional agents, they have to perform the automatic transfer of the production data to the control center.

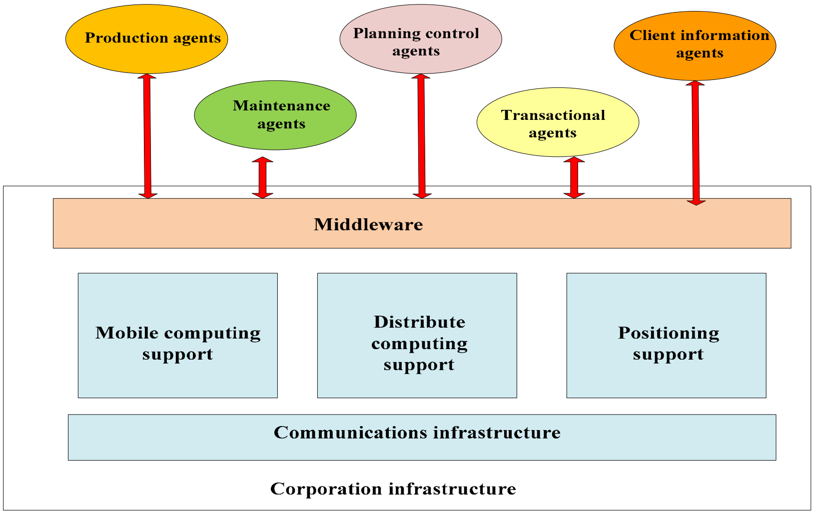


Fig. 2 Functional structure of the corporative system based on mobile agents

The middleware allows the autonomous and spontaneous data transfers between agents, making the data transference independently of the mobility degree and the locations of the agents. Additionally, generic communication channels are provided by the middleware in order to achieve transparency of the communications infrastructure used in each data transfer.

3 Middleware Architecture

The middleware exposed in this paper has a two layered structure, figure 3, formed by an interface layer with the mobile agents, named communication interface, and a middleware kernel. The kernel has two modules named management module and communication module. First it is responsible of the synchronized access and the refreshing of the data context used by the middleware and mobile agents. The second contains the implementation of all the primitives that allows to communication interface to give communication services.

The middleware facilitate the automatic and spontaneous data transfers between agents, making the data transference independently of the mobility degree and the locations of the agents. These data transferences are made by two communication modes: direct communication and communication by mailboxes. To choose the mode of communication, the middleware uses the priority of the agent. When the agent is high priority, the direct communications channel is chosen using the proper network infrastructure to fulfill real time requirement. However, when the communications have not priority, the middleware uses a communication scheme based on mailboxes where first the data are stored in mailbox and later they are sent. This mailboxes system is a distributed communication system specially designed for mobile agents, where the middleware does not allow migration of

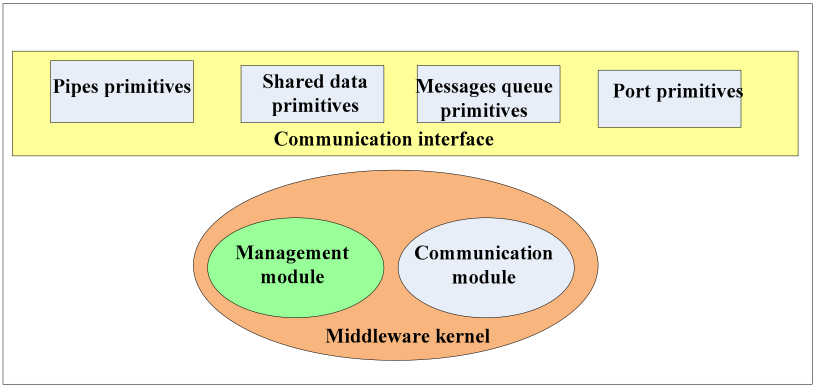


Fig. 3 Middleware structure

mailboxes of the agents because of this is the more suitable option for small and medium size networks; additionally the middleware delivers the messages using the *push* method, which is the most suitable in a real time context. Therefore, we don't need to synchronize mailboxes and delivery of messages.

The control of the data transferences is made using rules that are applied in automatic and unsupervised way by the middleware run-time agents. These rules are defined using the context data, specifying actions to execute and the values of the context data that triggers the actions. The agents apply a set of context sensitive rules that trigger a data flow between them through generic communications channels that are transparent to the network infrastructure used. A rule is composed of the specification of the condition as a trigger and the action to launch. The specification of the condition is based on context data that is combined using logical or relational operations and compared with constant values or the value of another context entity. The actions can be of two types: the execution of a program or the transfer of data using the generic communications channel provided by the middleware. To achieve the data communications, each agent must define the following elements: Declaration of the context data used by the agent, a set of events that are going to be controlled by rules and a set of configuration values such as names of the context entities, sampling period of the context data, etc.

This component provides the generic communication channels to the mobile agents, allowing the communications regardless of the type of infrastructure used. This layer plays a role of adaptor to the communication infrastructure, achieving the access to ubiquitous network and sending and receiving the data using the primitives of this layer. Currently, there are four types of communication primitives. The fist type, for agents local communication using pipe channel *Send_tube(route,id_pipe)* and *Receive_tube(route id_pipe)*, alternatively using messages queue *Send_queue(id_queue)* and *Receive_queue(id_queue)* . The second kind, to access to the shared data spaces on synchronized way *mem_sem(entity).* The third type, to communicate real time alarms or warning it is the primitive *warn(message_type).* Finally, the fourth type provides generic sockets *Send_Socket(address id_port) and Receive_Socket (address id_por).*

5 A Case of Use

The middleware has been used by Global Salcai-Utinsa S.A. Company; it is a road public transport corporation which operates in Gran Canaria (Canary Islands, Spain). All the data produced by vehicles activity is handled by it. To illustrate the size and the complexity of this management, we can comment some general figures of the company activity: the number of buses of the fleet is 307, it transports more than 30.000.000 of passengers yearly and it makes about 3.500 expeditions a day. This activity generates more than 180.000 data transactions per day, all of them automatically processed by the system with a lost data ratio bellow of 0.1%. The middleware uses a ubiquitous network integrated by three infrastructures. The first is a radio truncking system used for long distance communications to transmit high priority data associated with technical alarms, planning exceptions and events that can affect the normal operation of the fleet. The second is a wireless local network infrastructure based on IEEE 802.11 specification, that is used to transmit high volume of data to or from the mobile agents aboard, additionally it is needed to support the administration and maintenance of the agents aboard (hardware and software). Finally, the third infrastructure is formed by the ad-hoc wireless networks supported by small devices that are used for short distance communications on the vehicles; these are required to support the electronic payment by intelligent smart cards and to control vehicles combustible consumption.

6 Conclusions

In this paper a middleware for mobile agents has been explained, describing how the technological advances can help to the corporations, which operate in a mobile scenario, to improve its activities. Given the current state of information technologies, the work has demonstrated how the pervasive model and more specifically the middleware principles of operations can be applied to build information systems where mobile agents are integrated properly. Basically, the middleware permits to mobile agents a intelligent management of the communication infrastructures formed by different kind of network infrastructure, such as: public long distance network (for example GSM, UTMS or radio system), mobile local network (IEEE 802.11) and mobile short distance network (for example Bluetooth). The practical result is the improvement of the quality services of the corporation with attractive communication costs, playing a main role the middleware explained. Besides, it will allow the developing of flexible and scalable information system based on mobile services.

References

1. Román, M., et al.: GAIA: A Middleware Infrastructure to Enable Active Spaces. In: Mattern, F., Naghshineh, M. (eds.) PERVASIVE 2002. LNCS, vol. 2414, pp. 6–28. Springer, Heidelberg (2002)
2. Trevor, J., Hilbert, D.: AnySpot: Pervasive Document Access and Sharing. IEEE Pervasive computing, mobile and ubiquitous systems 6(3), 76–84 (2007)
3. Arikawa, M., et al.: NAVITAME: Supporting Pedestrian Navigation in the Real World. IEEE Pervasive computing, mobile and ubiquitous systems 6(3), 21–29 (2007)
4. Barretto, S., Mira, M.: Using Integration Frameworks for Developing Context-Aware Application. In: Markopoulos, P., Eggen, B., Aarts, E., Crowley, J.L. (eds.) EUSAI 2004. LNCS, vol. 3295, pp. 100–111. Springer, Heidelberg (2004)
5. Ranganathan, A., Cambell, R.: A Middleware for Context-Aware Agents in Ubiquitous Computing Environments. In: Endler, M., Schmidt, D.C. (eds.) Middleware 2003. LNCS, vol. 2672, pp. 141–161. Springer, Heidelberg (2003)
6. Murphy, A., et al.: LIME: A Middleware for Physical and Logical Mobility. In: IEEE 21st International Conference Distributed Computing Systems, pp. 524–533 (2001)
7. Mascolo, C., et al.: XMIDDLE: A Data-Sharing Middleware for Mobile Computing. Wireless Personal Communications 21(1), 77–103 (2002)
8. Yau, S., et al.: Reconfigurable Context-Sensitive Middleware for Pervasive Computing. IEEE Pervasive computing, mobile and ubiquitous systems 1(3), 33–40 (2002)

A Multiagent System for Efficient Portfolio Management

Vivian F. López, Noel Alonso, Luis Alonso, and María N. Moreno

Abstract. In this work we present a multiagent system to draw up an optimum portfolio. By using a distributed architecture, the agents are trained to follow different investing strategies in order to optimize their portfolios to automate the one year forecast of a portfolio's payoff and risk. The system allows to adopt a strategy that ensures a high rate of return at a minimum risk. The use of neural networks provides an interesting alternative decisions to the statistical classifier. With a modular agent composed by a few trained neural networks, the system makes investment decisions according to the assigned investment strategy and the behavior of the prices in a one-year period. The agent can take a decision on the purchase or sale of a given asset.

1 Introduction

The portfolio selection problem has received considerable attention in the financial literature [3, 5]. Currently, portfolio analysis can be approached from two points of view. First, we have portfolio selection, which Harry Markowitz introduced in 1952 [20]. The second aspect is portfolio management aimed at finding the optimal structure. Today, financial market problems are often solved using artificial intelligence. Despite the great deal of effort already put into making financial time series predictions [17], support vector machines [10], neural networks [24], prediction rules [8] and genetic algorithms [2], the prediction of a stock market index is still difficult to attain. The main reason for the complexity of this task is the lack of autocorrelation of index value which changes even in a one-day period.

The investment decisions challenge the investor's entire range of knowledge, that is always complicated, but particularly nowadays, due to advances in technology,

Vivian F. López · Noel Alonso · Luis Alonso · María N. Moreno
Departamento Informática y Automática, University of Salamanca, Plaza de la Merced S/N, 37008. Salamanca
e-mail: {vivian,noelalonso1,lalonso,mmg}@usal.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 53–60.
springerlink.com

diverse and voluminous information is becoming available to decision makers. Such is the case of portfolio management, that is still performed as craftwork. Selection is made according to the investor's favourite assets, or following the manager's ratings according to her experience, knowledge or intuition, but seldom based on formal grounds. This means that investors maintain inefficient portfolios that are not adjusted to the expected risk-payoff ratio.

In [18] we used a single system for efficient portfolio management, but we think that a multiagent system is more appropriate. The problem of advice personalized for new investors can be improved by means of a multiagent system. It is most natural for portfolio selection and monitoring because the multiple threads of control are a natural match for the distributed and ever-changing nature of the underlying sources of data and news that affect higher level decision-making processes [7]. Multiagent portfolio management with collaborative agents can improve the performance of a single portfolio management algorithm; because the agents in the cooperative search take into account the success of portfolio strategies followed by other agents. This paper focuses on the design of the portfolio management. According to Markowitz [21], the selection is grounded in the simple observation of prices that maximize the expected payoff at a given level of risk. Although the information is growing day by day, its in-depth processing is very complicated and not within easy reach of the average investor, who is usually unable to capture and interpret the data. In this work we have developed a collection of multiple agents to make an automatic data survey to draw up an optimum portfolio, to estimate the market risk and, at a given moment, to help the decision process regarding an asset, as well as anticipate user's information needs. The main objectives of the multiagent system are:

1. To automate the one year forecast of a portfolio's payoff and risk, showing the advantages of using techniques for developing distributed adaptative collections of information agents in portfolio management. It adopts a strategy that ensures a high rate of return at minimum risk.
2. To make the correct decision in the purchase or sale of a given asset, using a modular agent composed by a few neural networks to make investment decisions, classifying the process into three choices: buy, sell or do nothing.

2 Related Work

In this section we briefly review relevant research for the work presented: The current problem of portfolio selection is a stochastic online decision problem [15]. Cooperative search has been applied to hard computational problems in artificial intelligence with agents that have diverse search heuristics [4, 14]. [16] compares the performance of a system of many cooperative agents with simple search heuristics to a system of a few agents with more complex search heuristics, and [1] describe a cooperative search technique called "Go with the Winners Algorithms". Recent models of multiagent learning within artificial intelligence provide a hierarchy of agent models and allow strategic learning [11, 28, 25], where agents take advantage of models of the learning of other agents. In [7] techniques for developing distributed adaptive collections on intelligent information agents in the financial portfolio management are presented. The agents in [22] follow the framework of model-free learning: the agents in the model do not maintain an explicit model of the stock market. The current portfolio of an agent represents its cumulative learning, and it makes a small adjustment to its portfolio every time it observes new stock prices [13]. Another interesting proposal in that direction is the Multilayered Multi Agent Situated System (MMASS), defining a formal and computational framework relying on a layered environmental abstraction [6]. MMASS was related to the simulation of artificial societies and social phenomena, for which the physical layers of the environment were also virtual spatial abstractions. In the last decade, a number of other field-based approaches were introduced like Gradient Routing (GRAd), Directed Diffusion, ”Co-Fields” at TOTA Programming Model, CONRO etc [19]. In this direction there is a new approach [23] for designing an intelligent agent composed by a trained neural network individually specialized to make investment decisions according to the assigned investment strategy in the different markets.

3 Multiagent System Description

Based on this experience, a new multiagent model was developed for the automatic efficient management of investment fund portfolios that takes into account the history over a given period, adapts to dynamic market conditions and upgrades itself via the web periodically. The arquitecture of the multiagent system coordinates three kinds of agents [7]: Interface, Analysis, and Information agents, as shown in Figure 1.

In the system each investor is represented as an agent that can keep the investment for a certain amount of time according to individual preferences. Interface Agents interact with the user receiving user specifications and delivering results, in our model each agent distributes their wealth and divide it among the agents to learn its investment strategy from the sample data. Analysis Agent makes autonomous investment decisions in each investment period. Accordingly, each agent is optimized for the best possible forecasting of asset analysis in order to place profitable and risk adjusted decisions. The portfolio manager agent is targeted to find the optimum portfolio (optimal risk-profit on the efficient frontier) according to Markowitz theory [21]. Based on this knowledge, the Information Agents, decide what information is needed and initiate collaborative searches with other agents. In addition the Analysis Agent suggests the investor what assets to buy and the amount to be invested in each one to obtain a bigger payoff and a lower risk. Besides that it must indicate what is most suitable for the asset according to the daily evolution of prices and payoffs of each asset: keep, sell or buy.

3.1 The Interface Agent

The multiagent system can work with any portfolio on the stock-exchange market, each agent is forming its portfolio, so there are as many possibilities as assets. The

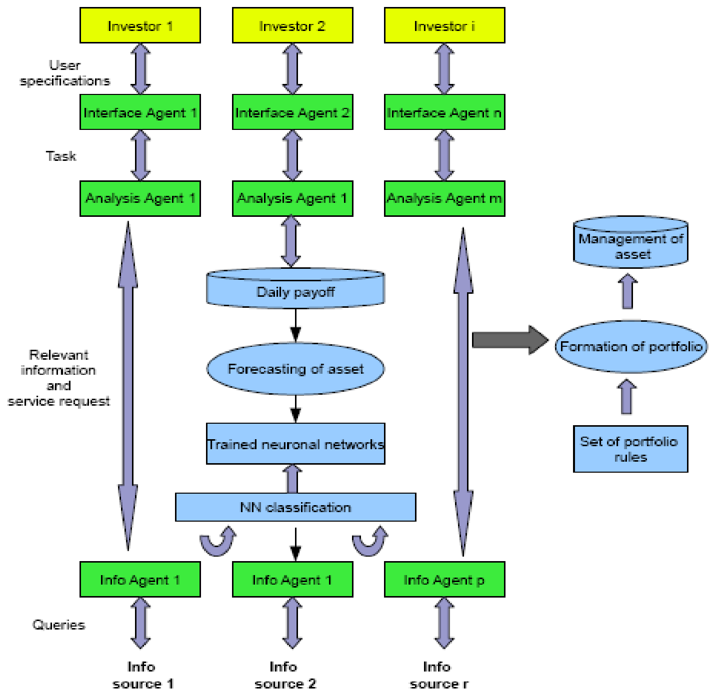


Fig. 1 The arquitecture of the multiagent system

users can have different roles: system manager, investor and financial adviser. The Interface Agent interacts with the users receiving user specifications, learning user profile information, collecting user information. The Interface Agent can be configured to make alerts of assets values. The alerts are sent by e-mail, being generated when the profitability of the portfolio overcomes certain limits.

The Interface Agent displays the standard information about a created users portfolio by the Analysis Agent and the portfolio evolution. In addition it allows the user to see his portfolios to buy and sell assets, and displays recent pricing and news information. The Analysis agent must always study it as a source of maximum available information and thus take decisions of buying or selling the asset.

3.2 *The Analysis Agent*

This agent allows to create efficient portfolios, interacting with the Information Agent and receiving user specifications from Interface Agent. The portfolio will be calculated from a series of assets and restrictions with better relation payoff/risk. At the same time it computes the Valuation of the Risk (VaR) by the Method Normal Delta [26]. This agent covers a variety of tasks such as the analysis of assets through estimation of expected payoffs, variances and covariances. The historical payoff is computed for each asset for a period of 321 working days, and the Analysis Agent obtains the following values: daily payoff (with respect to previous day), daily volatility (standard deviation), average daily payoff, daily profit or loss and VaR for each asset.

With the results obtained in the historical payoff phase, minimum variance point (MVP) is determined inside the boundary of production possibilities. Finally the MRP/MVP ratio is computed (maximum slope of the straight line) to maximize the payoff/risk ratio or, equivalently, maximum payoff at a minimum risk. In summary, the agent tries to find the right weights for each one of the portfolio components so that the agent can choose the best distribution. Once the payoff and risk are calculated, it selects the efficient portfolio and computes the VaR.

The Analysis Agent must simply study the evolutions of prices that are formed. It is composed by a few trained neural networks to make investment decision according to the different investing strategy. For every asset it trains a perceptron neural network with a single hidden layer [12]. For this case the significant input parameters are the value of daily participation, payoff and daily payoff. With them the net is trained to learn the behaviour of the prices in a one-year period, classifying them into three classes according to their daily profitability: Class 0 (do nothing), Class 1 (sell), Class 2 (buy). Each agent according to the investment strategy is classified for the best possible forecasting of asset.

In the training phase the agent uses 70 % of the available information and the remaining 30 % is used for the validation. The net has three input neurons, corresponding to the significant input attributes and three output neurons (classes). The number of neurons in the hidden layer is a parameter to play with to achieve a trade-off between efficiency and speed of training. For this agent, with three neurons an acceptable result is reached.

The net is trained using the Weka tool [27]. After training, we verify that the number of examples correctly classified depends on the fund in question, ranging between 96 % and 100 %, as we show in the results of the experiments.

We can generate a chosen number of agents for each investing strategy and we observe that the agent correctly classified the validation pattern. Once the net has been trained with the prices and final earnings, it can be consulted with any other input value in future periods, and they will be classified to help in the decision making on an asset. The agents aren't static since the strategy can change during the investment time horizon.

3.3 Information Agent

The Information Agent has models of information sources and information access strategies. It interacts with the Analysis Agent to return the user information via the Interface Agent. This agent retrieves, filters and fuses information relevant to the user, and anticipate user's information needs [7]. For example, to perform the prediction computations, for each asset in the portfolio, a data base with historical market information is defined with the following fields: ISIN code (fund registration), name of the asset, estimated time in portfolio (inversely dependent on risk), date of portfolio participation, value of participation, payoff, equivalent yearly rate (APR), observed volatility, market distribution and number of days to take into account.

In summary, the agent obtains the financial information by request of other agents or by his own initiative. It detects changes in the associated values for the asset and notifies these changes to rest of the agents.

4 Experiments

Experimentally, the model is tested in the Fibanc Mediulanum Banking Group Platform All Funds [9]. The data sets for performing the forecasting study of profitability and risk in a portfolio of values uses the 14 funds of different managers of the above mentioned values. The number of days to bear in mind is determined by the least amount of all the observations of each one of the 14 founds. For every type of agent historical payoff of the asset in a fixed period of time was computed.

For the classification of the behaviour of the prices, we performed two fundamental experiments:

1. Training a neural network for every fund.
2. An agent with the information of all the funds in the same period used in the analysis of the portfolio.

The worst results on the number of examples classified correctly were obtained by the agent that included all the funds for the analyzed period, that could only correctly classify 95.24 % of the cases. With one different agent for each fund the results range from 96.90 % corresponding to the Fidelity Fund up to 100 % of the majority, as can be seen in Table 1.

We have been investigating techniques to develop the Analysis Agent. The results obtained by means of neural networks were contrasted with those derived from a statistical method. Several approaches were considered based on statistical time series processing and curve adjustments. Results were poor so our conclusion was to

Table 1 Examples correctly classified for each asset

Name_Assets	Precision	Name_Assets	Precision
Franklin H.Y. ”A”	100.00	Dexia eq l aust ”c”	98.25
Dws Invest Bric Plus	100.00	Ubam Us Equi Value A	100.00
Aberdeen Asia Pac ”A”	97.90	Sch Eur Dyn Grwth A	100.00
Fortis l Eq Turk ”C”	100.00	Newton Hig Inc	100.00
Cre Suis.Cap Jp ”H”	98.30	Ing(l)inv Eur h.d ”x”	100.00
Challenge Country Mix (S)	98.60	Challenge Financial Fund	100.00
Challenge Germany Equity	97.30	Fidelity Eur S.C. ”E”	96.90

use nonparametric approaches, like neural networks. The classification errors with neural networks were much better in all the cases.

5 Conclusions

We compared a model of a single system for efficient portfolio management with a model of cooperative multiagent. Multiagent portfolio management can improve the performance of a single portfolio management algorithm; because the agents in the cooperative search take into account the success of portfolio strategies followed by other agents.

The model offers a methodology for the composition of efficient portfolios, becoming a more appropriate tool for investment decision making. The coordinated work of three agents (Interface, Analysis and Information) allows the system to adapt itself to new trends, since it keeps training with new information, so it can therefore adapt to dynamic market conditions taking into account the good results of previous periods.

The financial adviser agent, according to the type of investor (risk adverse, average risk or risk lover), can offer a scale of portfolios with a certain yield, in view of risk level. The multiagent system is able to suggest the asset the investor should buy and the time that it must remain in the portfolio to be profitable and adopting a strategy that guarantees high profitability and minimal risk for the investor, without restriction in the number and types of assets.

As a consequence of the distributed architecture, the Analysis Agent takes advantages of the parallelism in the different levels to make investment decisions according to the assigned investment strategy and the prices behavior. This agent can take a decision on the purchase or sale of a given asset. With the information of all the funds in the same period used in the analysis of the portfolio this management is more efficient and achieves better results than a single agent.

References

1. Aldous, D., Vazirani, U.: Go with the Winners Algorithms. In: Proc. 35th Symp. Found. Comp. Sci., pp. 492–501 (1994)
2. Allen, F., Karjalaine, R.: Using Genetic Algorithms to Find Technical Trading Rules. Journal of Financial Economics, 245–271 (1999)
3. Campbell, J.Y., Lo, A.W., MacKinlay, A.C.: The Econometrics of Financial Markets. Princeton Univ. Press, Princeton (1997)
4. Clearwater, S.H., Huberman, B.A., Hogg, T.: Cooperative Solution of Constraint Satisfaction Problems. Science 254, 1181–1183 (1991)
5. Cover, T.M.: Universal Portfolios. Math. Finance 1(1), 1–29 (1991)
6. De Paoli, F., Vizzari, G.: Context dependent management of field diffusion: an experimental framework. In: Proceedings of the Workshop from Object to Agents, WOA, Piagora Editrice, Cagliari, Italy (2003)
7. Decker, K., Sycara, K., Zeng, D.: Designing a multi-agent portfolio management system. In: Proceedings of the AAAI Workshop on Internet Information Systems (1996)

8. Dempster, M.: Computational Learning Techniques for Intraday fx Trading Using Popular Technical Indicators. IEEE Transaction on Neural Networks, 744–754 (2001)
9. Fibanc Mediulanum Banking Group, http://www.fibanc.es
10. Gestel, T., Suykens, J.: Financial Times Series Prediction Using Least Squares Support Vector Machines Within the Evidence Framework. IEEE Transactions on Neural Networks, 809–820 (2001)
11. Gmytrasiewicz, P.J., Durfee, E.H.: A Rigorous, Operational Formalization of Recursive Modeling. In: Proc. 1st Int. Conf. on Multi-Agent Systems (ICMAS 1995), pp. 125–132. AAAI Press, CA (1995)
12. Haykin, S.: Neural Networks: A Comprehensive Foundation. Prentice Hall, Englewood Cliffs (1999)
13. Helmbold, D.P., Schapire, R.E., Singer, Y., Warmuth, M.K.: On-Line Portfolio Selection Using Multiplicative Updates. Math. Finance 8(4), 155–177 (1998)
14. Hoog, T., Williams, C.P.: Solving the Really Hard Problems With Cooperative Search. In: Proc. 11th Nat. Conf. on Artificial Intelligence (AAAI 1993), pp. 231–236 (1993)
15. Irani, S., Karlin, A.R.: Online Computation. In: Hochbaum, D.S. (ed.) Approximation Algorithms for NP-Hard Problems, ch. 13, pp. 521–564. PWS Publishing (1997)
16. Knight, K.: Are Many Reactive Agents Better Than a Few Deliberative Ones? In: Proc. 13th Int. Joint. Conf. on Artificial Intelligence. IJCAI 1993, pp. 432–437. AAAI Press, CA (1993)
17. Kodogiannis, V., Lolis, A.: Forecasting Financial Times Series Using Neural Network and Fuzzy System-Based Techniques Neural Computing & Applications, pp. 90–102 (2002)
18. López, V.F., Alonso, L., Moreno, M.N., Segrera, S., Belloso, A.: A System for Efficient Portfolio Management. In: Yin, H., Tino, P., Corchado, E., Byrne, W., Yao, X. (eds.) IDEAL 2007. LNCS, vol. 4881, pp. 980–989. Springer, Heidelberg (2007)
19. Mamei, M., Zambonelli, F.: Field-Based Coordination for Pervasive Multiagent Systems, pp. 241–248. Springer, Heidelberg (2006)
20. Markowitz, H.: Portfolio Selection. Journal of Finance 7 (1952)
21. Markowitz, H.M.: Portfolio Selection: Efficient diversification of investments. Blackwell Publishers, Malden (1991)
22. Parkes, D., Huberman, B.: Multiagent Cooperative Search for Portfolio Selection. Games and Economic Behavior 35, 124–165 (2001)
23. Plikynas, P.: Portfolio design and optimization using neural network based multiagent system of investing agents. In: Sakalauskas, L., Weber, G.W., Zavadskas, E.K. (eds.) International Conference 20th EURO Mini Conference Continuous Optimization and Knowledge-Based Technologies (EurOPT 2008), EUROPT 2008 Selected papers, Neringa, LITHUANIA, Vilnius, pp. 137–142 (2008)
24. Tino, P., Schittenkopf, C.: Financial Volatility Trading Using Recurrent Neural Networks. IEEE Transactions on Neural Networks, 865–874 (2001)
25. Vidal, J.M., Durfee, E.H.: Learning Nested Agent Models in an Information Economy. J. Exp. Theoretical Artificial Intelligence 10, 291–308 (1998)
26. Vilariño, A.: Tubulencias Financieras y Riesgos de Mercado. Prentice Hall, Madrid (2001)
27. WEKA, http://www.cs.waikato.ac.nz/~ml/weka/index.html
28. Wellman, M.P., Hu, J.: Conjectural Equilibrium in Multiagent Learning. Machine Learning 33, 179–200 (1998)

Managing Real-Time Web Services through Agents

Elena del Val, Martí Navarro, Vicente Julián, and Miguel Rebollo

Abstract. Time is an important Quality of Service (QoS) parameter in services. In many situations, the response provided by a service could be completely useless if it is not provided on time. In this paper the infrastructure to provide a real-time web service (RTWS) is described. These RTWS are provided by agents with the capability of negotiation and guaranteeing the service execution time. These kind of agents control the service execution time and allow a provider-client negotiation process in order to arrive at an agreement on when the service response is it to be provided. If an agreement is reached, these agents also guarantee that the agreement is going to be fulfilled. Finally, tests to validate RTWSs and the behavior of provider agents are presented.

1 Introduction

Nowadays, service-oriented computing (SOC) brings additional considerations, such as the necessity of modelling autonomous and heterogeneous components in uncertain and dynamic environments. Such components must be autonomously reactive and proactive yet able to interact flexibly with other components and environments. As a result, they are best thought of as agents who collectively form Multi-Agent Systems. SOC represents an emerging class of approaches with Multi-Agent System-like characteristics for developing systems in large-scale open environments. For this reason agent orientation is considered an appropriate design paradigm to act as providers of services. Agents due to their characteristics can offer more flexibility in interactions between services and clients. Furthermore, agents through negotiation or agreement protocols can provide more suitable services attending to quality of service (QoS) client requirements.

Elena del Val · Martí Navarro · Vicente Julián · Miguel Rebollo
Technical University of Valencia, Camino de Vera s/n. 46022 Valencia, España
e-mail: {edelval,mnavarro,vinglada,mrebollo}@dsic.upv.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 61–68.
springerlink.com

The QoS offered by web services is becoming the highest priority for service providers and their partners due to the proliferation of web services as a solution to application integration.Service consumers like to obtain guarantees related to the QoS of the required services. Whether service providers can offer and fulfill these guarantees depends on the availability of the required resources at that time. Service execution time is one of the most important QoS parameter because some services could be completely useless if it are not provided on time. For many control systems such as manufacturing systems, video and audio systems or factory controllers, timing behavior is an important aspect. Currently, the majority of the proposals related to time and web services deals only with the specification of some temporal properties at a description level. Despite the fact that these proposals take into account time properties and constraints, none of them provide mechanisms to guarantee them at execution time.

This paper presents an architecture which offers real-time web services and guarantees their service execution time through a service provider agent. This agent controls the execution of the real-time web services, analyzes if a request can be provided by a service before a deadline and also schedules the service execution in order to guarantee the resources needed for the service.

The paper is organized into the following sections: in Section 2, our proposal of architecture to support real-time web services is presented. In Section 3, a set of test configurations is presented in order to validate the behavior of the real-time web services and also to evaluate the benefits of the architecture. Finally, in Section 4, conclusions and future work are described.

2 System Architecture to Support Real-Time Web Services

In this section an architecture based on real-time web services and agents is presented. This architecture deals with the requirements to consider service execution time. These requirements are: the provision of service execution time information at the level of the individual service, plus the design of systems aware of execution time, whilst ensuring that an acknowledged executed time is actually provided by the service. The main components of the architecture are (see Figure 1):

- **Real-Time Web Service (RTWS):** It is characterized by two non-functional parameters: service execution time and the probability of finishing by this time.
- **Service Provider Agent (SPA):** This agent offers real-time web services and when it receives a client's request can analyze if it has enough resources and time to execute the service and control the service execution.
- **Execution Platform:** In order to develop and execute RTWS, a Real-Time Operating System (RTOS) is necessary. A RTOS is a multitasking operating system intended for real-time applications. RTOS provides facilities which, if used properly, guarantee deadlines which can be met generally (soft real-time) or deterministically (hard real-time).

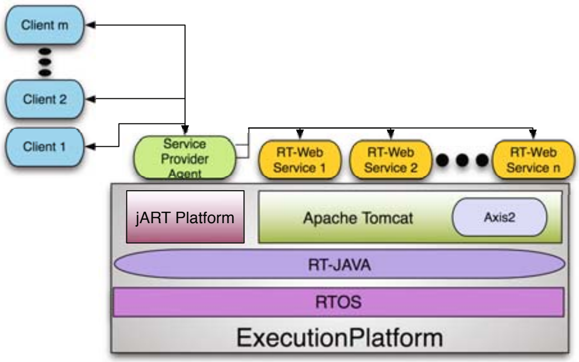


Fig. 1 Architecture for Time-bounded Services

In the next subsections the components of the system architecture are explained with more details.

2.1 Real-Time Web Services

Sometimes, the concept of real-time generates some confusion. In some situations, real-time applications are interpreted as on-line applications, but in our context, real-time applications means an action that answers to an external event in a timely and predictable manner. Therefore, we define a Real-Time Web Service (RTWS) as a service in which a bound on the response time is known and, to be more realistic, it has a probability associated to it in order to model *soft deadlines*. For this reason, the service description, apart from the usual information related to Inputs, Outputs, Preconditions and Effects (IOPEs), should contain an explicit representation of time and probability as non-functional parameters. These parameters facilitate the client's selection task between a set of services that offer the same functionality.

In order to offer this information to the clients, an OWL-S service description, extended with non-functional parameters, has been used (see Figure 2). This description introduces two non-functional parameters which represent service execution time and probability. The value of the service execution time is represented by an average execution time. This execution time is obtained by measuring the cost of the service off-line.

The description of a RTWS adds new features that change the usual service description. These changes not only affect the service description but also the service implementation. Although Java is the language most widely used to develop web-services, the use of Java for developing time-bounded services is inadequate for several reasons: the use of *Garbage Collector*, the dynamic load of the classes or problems with priority inversion. Therefore, it is not possible to apply a standard Java runtime environment for RTWS. In order to solve this problem, there is an extension to adapt Java to real-time applications. This extension is the *Real-Time Specification for Java* (RTSJ). RTSJ is an extension of the Java language for

```
<process:AtomicProcess rdf:ID="ServiceOneProcess">
...
<time:hasLocal>
  <time:time-Expression rdf:ID="S-Time">
  <expr:expressionBody rdf:datatype="http://...#long">
  23
  </expr:expressionBody>
  </time:time-Expression>
</time:hasLocal>
<probability:hasLocal>
  <probability:probability-Expression rdf:ID="S-Probability">
  <expr:expressionBody rdf:datatype="http://...#float">
  0.83
  </expr:expressionBody>
  </probability:probability-Expression>
</probability:hasLocal>
```

Fig. 2 Non-functional parameters in an OWL-S real-time service description

developing RTWS over Real-Time Operating Systems without modifying the standard. The most important improvements of the RTSJ are: thread scheduling, memory management, synchronization, asynchronous events, asynchronous flow of control, thread termination and physical memory access.

2.2 Service Provider Agent

The SPA has the capability to analyze and control the service execution, and more important, to establish an agreement with the client to guarantee that the selected service would be executed before the client deadline. Therefore, the main functions of the SPA are:

- to analyze if a service can be executed before a maximum time
- to establish an agreement with the client
- to execute RTWS and check that the services are provided on time in order to fulfill the agreement

In order to guarantee the correct functionality of the system, the implementation and execution of the SPA has been using the jART platform [1]. This platform is specially developed to offer the necessary mechanism to implement SPAs using RT-Java as programing language, and to control and execute the real-time tasks of the SPA's. The functionality of the SPA is described with detail in the following subsections.

2.2.1 Schedulability Analysis

The SPA must decide if it can undertake to provide a requested RTWS before a deadline established by the client. To do so, the SPA must analyze if the RTWS will have the necessary resources (memory, CPU, shared resources) to provide the service before the deadline expires. The task of analyzing whether a RTWS can be executed on time is a relatively easy one, using well-known scheduling techniques

[2][3] as long as the service execution time is known. Otherwise, prior to performing the analysis, it is necessary to make an estimation of the temporal cost of the RTWS execution. In this case, the SPA knows the service execution and uses an analyzer to check if a service can be provided before a deadline. If the service cannot be provided before that deadline, the analyzer proposes a new deadline. Furthermore, the SPA reserves the necessary resources to execute the service until the SPA receives an answer from the client accepting or rejecting the proposed deadline. The aim of the reservation is to guarantee the service execution if the client finally accepts the proposal.

2.2.2 Agreement Establishment

The interaction between the clients and the SPA is made through an agreement protocol. The agreement is based on the standard WS-Agreement [4] and using the API provided by the Fraunhofer Institute SCAI[1]. In our arquitecture, the temporal cost of communications between the SPA and the client through the network is predictable and it is considered in the different analysis to guarantee an answer before certain timeout. Unfortunately, this assumption is only true for specific networks, for example CAN networks [5]. Thus, if this architecture would be extrapolated to common network media (Ethernet, serial, wifi, etc...), this feature would be lost. The interactions of the agreement protocol between clients and SPA are as follows (see Figure 3):

- The SPA publishes the templates of the RTWS provided by the platform. These templates contain the terms that are going to take part in the agreement. In this case, the terms are the RTWS features: service execution time and its associated probability.
- A client queries the agreement templates from the SPA. Based on a suitable template, the client analyzes the service execution time t and, considering the value of this term, creates a new proposal providing the desired deadline ($t_{deadline}$) in absolute time (date and hour). Before this deadline the service should finish its execution. The offer is sent to the SPA. The client waits the answer of the SPA until a timeout (t_{wait1}). If the client does not receive an answer, the client cancel the agreement protocol.
- The SPA analyzes the client's template. This process is temporal bounded. Basically, the SPA checks if the RTWS requested by the client is available and if it can be provided before the client deadline ($t_{now}+t \leq t_{deadline}$). If the service can be provided on time, the SPA returns the same deadline. Otherwise, the SPA proposes a new deadline $t_{deadlineNew}$. In both cases, the SPA also returns the probability of service execution success before the deadline p'. In order to guarantee that required RTWS is going to be provided on that deadline, the SPA schedules its service execution. The SPA waits an anwer from the client during a time (t_{wait2}). If the SPA does not receive an answer, the SPA removes the scheduled service execution.

[1] http://packcs-e0.scai.fraunhofer.de/wsag4j/

- If the client agrees with the proposed deadline $t_{deadlineNew}$ and its probability is higher than the client threshold ($p' \geq \alpha$), the client sends an agreement with the final values of the agreement terms to the SPA.

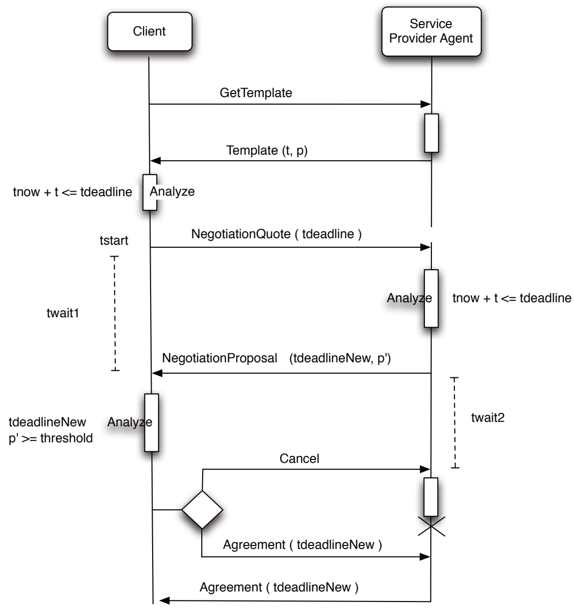


Fig. 3 Agreement protocol proposed to negotiate the execution time of the RTWS

2.2.3 Real-Time Web Service Execution and Control

Once the commitment has been established, the SPA indicates to the RTOS scheduler that the RTWS can be selected to be executed. The RTOS scheduler takes into acount the deadlines of the rest of the RTWSs to select which service must be executed. When the RTWS execution ends, the SPA checks if the execution was correct and if the deadline was accomplished according to the agreement established with the client.

3 Performance

To evaluate the benefits of providing RTWS through agreement protocols to negotiate deadlines some tests have been done. The test configuration consists of a set of clients located in a *hostA* and ten services situated in a *hostB* . Both hosts are in the same network. Client instances are launched for 2 minutes following exponential distribution in which the value of λ parameter is 0.3 (approximately one client every 3 seconds). Each client starts the agreement protocol with the SPA. The clients maximum and minimum deadlines are obtained following a random function which gives a minimum deadline between 60 and 70 seconds and a maximum deadline between 80 and 90. Each client also has a probability threshold in order to accept only those deadlines with a certain probability.

The SPA proposes to the client three types of deadlines (See formula 1):

- *The first client proposal*. This deadline indicates that the QoS response is high because it is a more accurate deadline considering client preferences.
- *A new deadline and the client accepts it*. This deadline indicates that the quality of the service response is lower than the first case because there is a little variance, but the deadline is also adecuate for the client's preferences. This quality is calculated considering the difference between the deadline provided by the SPA in the negotiation process(*NegDL*)and the deadline proposed by the client(*DL*). This value is divided by the difference between the higher (*MaxDL*) and the lower (*MinDL*) admisible client deadlines.
- *A new deadline and the client rejects it*. This indicates that the deadline provided is not in the range of admisible deadlines considering the client's preferences.

$$QoS(s) = \begin{cases} 1 & \text{(a)} \\ \frac{NegDL - DL}{MaxDL - MinDL} & \text{(b)} \\ 0 & \text{(c)} \end{cases} \tag{1}$$

The aim of the test is to evaluate the QoS provided by the SPA and the system CPU utilization considering the type of proposed deadline.

The results obtained after one hundred executions of this test are shown in Table 1. The first row is the average number of clients requests. The second row shows the average number of client requests that dealt with before the final deadline. The CPU utilization is shown in the third row. The sum of the quality of each service (QoS(s)) is divided by the total requests is shown in the fourth row. In both cases the CPU utilization is very similar, but the system utilization is higher using the agreement protocol because all of the service executions are scheduled in order to optimize the CPU utilization. The main difference between a SPA which has the capability of negotiation and other without this capability is the quality of the service response. With the SPA with negotiation the quality increases considerably. This is because the SPA, in the case that a client request cannot be provided before a deadline, does not directly reject the client request, but instead offers an alternative, so the client can decide whether to accept a new deadline to get the service response. Using the SPA with negotiation an alternative deadline is provided, although the quality of the service response is lower, but this is better than rejecting the client requests or not satisfying the clients expectations. Therefore, the use of the SPA with negotiation provides a system with better CPU perfomance and quality, due to the fact that the SPA does not refuse services that could be fulfilled with a little more time.

Table 1 Results with capacity of negotiation in the SPA

	With Negotiate	Without Negotiate	Gain (%)
$\bar{x}$ no. of total requests	32	39,66	-19,31
Fulfilled requests	15,66	6,66	135,13
CPU utilization	0.94	0.85	11,90
QoS	0.336	0.188	78,72

4 Conclusions and Future Work

In this paper, an architecture which offers real-time web services through provider agents (SPA) that follow an agreement protocol is presented. The SPA controls the execution of the real-time web services, analyzes if a request can be provided by a service before a deadline and also schedules the service execution in order to guarantee the resources needed for the service. This architecture is executed over a real-time operating system which provides mechanisms that guarantee deadlines. In order to validate the behavior of the architecture, several tests have been done. The tests show that the use of a SPA improves the quality of the executed services through a negotiation process with the clients and increases CPU utilization.

An extension of this architecture is planned in order to offer a longer negotiation time for agreements and also to offer time-bounded service compositions. Furthermore, to avoid a bottleneck in the SPA, there is a proposal to replicate , if necessary, the SPA and distribute the workload.

References

1. Navarro, M., Julián, V., Soler, J., Botti, V.: jart: A real-time multi-agent platform with rt-java. In: 3rd IWPAAMS, pp. 73–82 (2004)
2. Liu, C.L., Layland, J.W.: Scheduling algorithms for multiprogramming in a hard-real-time environment. J. ACM 20(1), 46–61 (1973)
3. Burns, A., Wellings, A.: Advanced fixed priority scheduling, pp. 32–65 (1996)
4. Grid Resource Allocation and Agreement Protocol WG (GRAAP-WG). Web services agreement specification (ws-agreement), `http://www.ogf.org/documents/GFD.107.pdf`
5. Controller area network can, an in-vehicle serial communication protocol. In: SAE Handbook 1992, pp. 20341–20355 (1990)

Open MAS Architecture. Providing Real Time Solutions

Martí Navarro, Sara Rodríguez, Vicente Julián, and Vivian F. López

Abstract. This presents a study in which a high level abstract architecture was used to design open multi-agent systems and virtual organizations that offer services with temporal constraints implemented by Real-Time Agents. The results will demonstrate how the proposed architecture, with features that make it suitable for development of open MAS (Multi-Agent Systems), allows us to add specific functionality and real time services.

Keywords: Open MAS , Real-Time Agents, Virtual Organizations.

1 Introduction

One of the goals of multi-agent systems is to construct systems capable of autonomous and flexible decision-making, and of cooperating with other systems within a "society". This "society" should take certain characteristics into account, such as distribution, constant evolution, or a flexibility that allows its members (agents) to enter and exit at will, a correct organizational structure, and the ability for the system to be deployed on different types of devices. Each of these characteristics can be achieved through the open multi-agent system and virtual organization paradigm. This paradigm was conceived as a solution to the management, coordination and control of the agents' behavior. Organizations should not only be able to describe the structural composition (i.e. functions, agent groups, interactive and relationship patterns between roles) and the functional behavior (i.e. agent tasks, plans or services), but should also be able to describe the behavioral norms

Sara Rodríguez · Vivian F. López
Departamento Informática y Automática Universidad de Salamanca
Plaza de la Merced s/n, 37008, Salamanca, Spain
e-mail: vivian@usal.es, srg@usal.es

Martí Navarro · Vicente Julián
Departamento Sistemas Informáticos y Computación. Universidad Politécnica de Valencia
46022, Valencia, Spain
e-mail: mnavarro@dsic.upv.es, vinglada@dsic.upv.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 69–76.
springerlink.com

for the agents, the entry and exit of dynamic components, and the formation, which is also dynamic, for the groups of agents. In general, it is necessary to define the standards and platforms required for the interoperability of the agents that meet these requirements. This article attempts to present a study in which a high level abstract architecture was applied with the specific intent of addressing the design of open multi-agent systems and virtual organizations using services with temporal constraints offer by Real-Time Agent. It will be shown how the proposed architecture, with its features that make it suitable for development of open MAS (Multi-Agent Systems), make it possible to add specific functionality and services for real time.

It has been necessary on many occasions to include specially designed agents within a society so that they can carry out particular services involving certain temporal restrictions when they are executed. These special agents, referred to as real time agents [5], are specifically designed to work in environments whose systems are characterized by strong temporal restrictions and require a specific infrastructure to function correctly and ensure that the restrictions are satisfied. It is further necessary to consider that in this type of environment, correcting the system depends not only on the logical computed result, but also on the moment at which the result is produced [10].

The article is structured as follows: section 2 describes an abstract architecture specifically adapted to work with open multi-agent systems and virtual organizations. Section 3 explains the principal characteristics of a real time system. Section 4 shows how the open architecture has been extended to support agents in real time; and finally some conclusions are given in section 5.

2 Ovamah

The primary concepts used in this study include the OVAMAH platform than can use agent technology in the development process and apply decomposition, abstraction and organization techniques; and the main features of temporal bounded services that are provided by Real-Time agents with the capability of guaranteeing the service execution time.

The proposed methodology used in this study uses the OVAMAH platform. OVAMAH is based on THOMAS (MeTHods, Techniques and Tools for Open Multi-Agent Systems) architecture [1] [3]. THOMAS is essentially formed by a set of services that are modularly structured. It uses the FIPA architecture, expanding its capabilities with respect to the design of the organization, while also expanding the services capacity. The information and mechanisms used by the traditional FIPA Directory Facilitator (DF) are insufficient to deal with open systems and system dynamics, having several limitations such as their very basic service descriptions (name, type, protocol, ontology, language, ownership and properties) and permissible functionalities (register, deregister, modify and search), which do not at any point take into consideration the virtual organizations. Another DF limitation concerns the service discovery, whose search algorithm does not use any semantic information and it does not consider service compositions. Given that Service Oriented Architectures have dealt with these aspects issues in depth, THOMAS proposes to incorporate this SOA vision into the way services are han-dled in an agent platform. Essentially, THOMAS updates the FIPA approach by presenting all the architecture functionalities specified as SOA services and updat-ing the Directory Facilitator (DF), that is, the FIPA module in charge of managing services, in order to use services specified according to SOA standards. This mod-ule, now called the Service Facilitator (SF), can not only register and offer affor-dability and discovery of services, but can also manage the service requests. Moreover, these services can be related not only to agents, but also to organiza-tional units. THOMAS also includes a new module called Organization Manage-ment Service (OMS) which is responsible for managing the lifecycle of virtual or-ganizations (in the same way that AMS deals with the lifecycle of agents) and for managing norms. More specifically, it controls how Virtual Organizations (or Or-ganizational Units) are created and the way entities participate inside such organi-zations (which entities participate, how they are related to each other and which roles they play). Another important module in THOMAS is the PK (Platform Ker-nel), which controls not only the communication but also the life cycle of internal agents. It is a classical standard FIPA abstract architecture PK module. All these modules can be seen, imbedded into the case study, in Figure 1.

3 Real-Time Agents and Temporal-Bounded Services

Sometimes, the concept of real-time generates confusion. In some situations, real-time applications are interpreted as on-line applications, but in the context of the present study, real-time applications refers to an action that answers to an external event in a timely and predictable manner.

There can exist various tasks within an organization that require some sort of temporal control in their execution. Because of their particular characteristics, these tasks cannot be performed by agents that do not possess the appropriate infrastructure. In order to ensure the correct execution, as per the temporal restrictions associated with these tasks, the tasks must be deployed on a real time operating system (RTOS).

The Real-Time agent model incorporated in the OVAMAH architecture is an agent that can offer Temporal Bounded Services and execute the different tasks associated with these services. The tasks are executed according to the temporal restrictions that define the maximum allowable time for the service to provide a result, and the temporal restrictions associated with each of the tasks that comprise the service. A Temporal Bounded Service (TBS) is considered a service that imposes a limit on the allowable response time. Therefore, apart from the usual information related to Inputs, Outputs, Preconditions and Effects, the service description should contain an explicit representation of time as non-functional parameters. With the use of a time parameter, it is possible to indicate how much time the service will need to complete its execution in the worst case scenario. In order to offer this information to the clients, an OWL-S service description, extended with non-functional parameter, has been used. The value of the service execution time is represented by an average execution time, which is obtained by measuring the cost of the service off-line. Additionally, preconditions and effects could contain modal expressions that make it possible to express relationships between different services and states. These relationships are expressed using an extension of the RuleML for modal logics [2].

In order for a real time agent to offer TBS, it must possess certain qualities such as:

(i) The ability to determine whether a TBS can be executed before a maximum time. The client will establish the start time for executing the TBS, the maxi-mum time to complete the TBS and, if the execution of the TBS must be peri-odic, the frequency of execution. These parameters will determine the temporal constraints of the TBS. The real time agent should include a planning compo-nent that can perform a viable analysis of the request. There are TBSs that, be-cause of their characteristics cannot be analyzed by the planning agent. These must then be implemented by the real time agent, which would otherwise not offer that particular TBS. There are certain schedules, for example, that can only manage periodical tasks. Thus a TBS that must be executed aperiodically cannot be analyzed by the real time agent.

(ii) The ability to execute TBSs and check that they are provided on time in order to fulfil the client's request.

As an additional option, the RTA can possess the ability to establish a relationship with clients that request a TBS through an agreement protocol, and negotiate the maximum allowable time to execute the TBS in the event that the RTA cannot execute the TBS within the time indicated by the client.

On occasion, the TBS service request cannot be met within the deadline. In this case, the service can offer the possibility of negotiating a new deadline with the client, based on the conditions analyzed by the RTA provider. To ensure that the negotiation is carried out, it is necessary for both the client and the provider to implement the agreement protocol shown in [8].

The following section explains how the OVAMAH architecture was expanded to include real time agents.

4 Extending the Architecture with Real-Time Agents

This section will show how the OMS and SF platform modules afford the possibility of managing open systems that permit the integration of real time agents and offer TBS. As a whole, these modules can be seen, as a platform-independent set of services in a framework that manages virtual organizations for open systems. This is known as the OVAMAH Framework. The OVAMAH framework provides a virtual organization in which any entity is automatically included, and a general role that allows the entity to ask for service descriptions so that it can fullfil its needs. Using a service description, the client can be informed about the roles required to fulfill the given request. The client can also provide a specific service within the organization. OVAMAH makes it is possible to incorporate agents that can openly provide the system with new services. This ability can be used to add agents to the platform, particularly those that can execute tasks and/or services that, because of their characteristics, must be executed according to certain temporal restrictions. The use of these agents in a real time environment is thus quite beneficial. OVAMAH provides a valid logic for modeling a real time MAS specification.

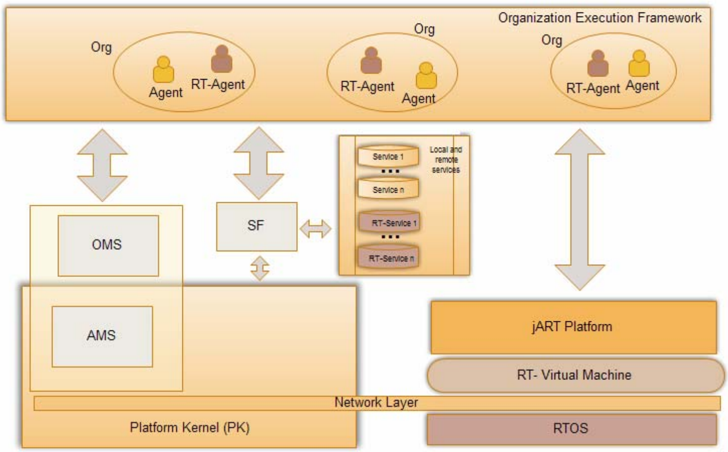


Fig. 1 OVAMAH Architecture for Time-bounded Services

One of the first factors to bear in mind is the need for a specific software to execute agents in real time. As mentioned in the previous section, in order to ensure that temporarily restricted tasks are carried out in time, they must be executed on a RTOS. To properly manage and control real time agents and their corresponding tasks, they must be executed on a platform that offers real time planning, temporally bounded communication between agents, and complete control over the processes executed in the system. One example of a system that offers these capabilities is the jART [9] real-time MAS platform, which can be easily integrated within the OVAMAH architecture s it follows the FIPA standard that was used to design the components of the OVAMAH architecture. The OMS in OVAMAH offers the services that are needed to ensure the correct functioning of an organization. These services can be dynamic or structural. The dynamic services can manage the dynamic entrance and exit of the agents in the system. The structural services are what make it possible to register or unregister roles, norms and units within the organization. The roles will have associated rules of interaction and behavior that allow us to define which type of services can be offered and required by an agent that is performing a specific role. The rules allow us to define restrictions, obligations and permission between agents at the moment they execute, request or implement services. The type of services available is controlled by the system through these norms. In our case, the norms require the TBS providers to be RTA, that these agents have a planner capable of analyzing the TBS, and furthermore, that they be able to control the execution of tasks associated with the TBS.

The system will specify the profile of the service providers by defining roles as well as the norms for processing service requests or the available results. This way, in the event of any illicit or improper behavior on behalf of the provider, the system will respond with sanctions. In this case study, the system will have a type of super-role RTA, which will be an "external" or accessible role that specializes in different types of sub-roles according to the type of planner used by the RTA. The following sub-roles will be available:

-RTA_Pd: The RTA agent role that can perform only periodical tasks, as it imple-ments a viability analyzer that can only analyze this kind of task. For example: monotonic rate algorithm [7] or earliest deadline first [7].

-RTA_Apd: The RTA agent role that can perform aperiodical roles using planners that can analyze them. For example: the slack stealing algorithm [6]. The planners capable of analyzing aperiodical tasks can also be used to analyze periodical tasks.

At the software level, OMS establishes a hierarchy of roles so that an agent with a role can offer services associated with roles hierarchically superior to its own, so long as this action is not contrary to the established norms. In this case, for example, an agent with the role of RTA_Pd can directly request services as-signed to the RTA role, but inversely the RTA agent must first request permission from the OMS to acquire the RTA_Pd role to use the services assigned to that role.

If we move to a hardware level, a unit that offers or includes TBS services, it should contain a real time agent that includes the following intrinsic characteristics: (i) It should be executed on a RTOS. (ii) It should have an appropriate planner available to carry out a viability analysis and plan tasks associated with the TBS:

The SF announces services that are required for the system to function properly. These services only have a Profile assigned to them, a structure for entrances/exits, preconditions/postconditions that it must comply with. The external agents can request the existing list of services and determine whether entering or not entering forms part of the organization, and under which roles. In our case, a RTA can request carrying out a TBS and it will be the platform responsible for confirming if the agent is capable of carrying out the specified type of service. If an that agent lacks the appropriate RTA characteristics attempts to provide a similar type of service, the platform will let it know the request has been rejected.

The type of services that can be offered is controlled by the system norms. All of the posible behaviors within the system will be ccontrolled by the norms defined with the following syntax:

`<norm>::=<deontic_concept>(<action>[<temporal_situation>][`**`IF`**` if_condition>])[`**`SANCTION`**`(<state>)] [`**`REWARD`**`(<state>)]` donde <deontic_concept>::= **OBLIGED | FORBIDDEN | PERMITTED**

and the action will be dialogical (send a message) or involving a request, provision or service registration (REQUEST | SERVE | REGISTER `<service>`). In `<if_condition>` it will be posible to indicate the results of the functions or services. The norms will reflect the necessary restrictions for real time tasks, including:

-The role that a RTA can perform is associated with the type of planning that it has implemented. An agent with a planning capacity limited to analyzing periodical tasks cannot perform the RTA_Apd role. However, an agent with a planning capacity that can analyze aperiodical tasks can, in fact, perform the RTA-Pd role.

```
FORBIDDEN RTA_Pd REQUEST AdquireRole(RTA_APd,GlobalUnit)
```

```
PERMITTED RTA_APd REQUEST AdquireRole(RTA_Pd,GlobalUnit)
```

-The RTA_Pd can only execute periodical services.

```
OBLIGED PK SERVE Suspend(RTA_Pd) IF RTA_pd REQUEST AdquireRole(RTA_APd,GlobalUnit)
```

-An agent that commits to carrying out a service is required to do so within a determined period of time, otherwise it will be sancioned within the organization.

```
OBLIGED PK SERVE Suspend(RTA) IF RTA_Serv_Profile (ServiceResult(requestTime ?time)) > "spentTime"
```

-An agent can delegate a TBS so long as the agent to which it assigns the service is an RTA that meets the necessary specifications.

```
PERMITTED RTA REQUEST TBS(TBSProfile) AFTER RTA REQUEST TBS1(TBS1Profile) AND Quantity(ServiceResult(TBS1,RTA,?Client, "completed")) > "1"
```

-If an agent requests a service from a RTA, the service cannot be carried out if either the allotted time has expired, and/or there are no available resources.

```
OBLIGED PK SERVE Suspend(RTA) IF RTA_Serv_Profile (input(requestTime ?time))>"currentTime"
```

All of the information related to existing units, their roles, service profiles and norms will be defined in the OWL-S file using an ontology and service organization. Upon initializing the system, the OMS will read the structural information from the file and create and initiate the organization. The OMS will internally save the list of norms that define the affected role, the norm´s content and the roles responsible for controlling if the norm is satisfied.

Finally, the SF will save the profiles of the services that are offered and/or needed by the system, as well as the relationship between the service provider entities and their corresponding *Process and Grounding* service During the grounding process it is possible to specify the specific parts of the real time systems that were not detailed by the norms, such as the type of planner that is used, the RTOS and/or multi-agent platform to use, as well as the entrance parameters that the service user must bear in mind when submitting a request (such as the moment to initiate the service, the frequency, and, when available, the corresponding priority and deadline.

5 Conclusions

The social factors that make it possible to arrive at a solution within multi-agent system organizations are also becoming increasingly important for structur-ing interactions within open and dynamic worlds. Any infrastructure that can sup-port the execution of multi-agent applications within these environments must be robust, efficient and adaptive over time. Given the characteristics of these open environments, particularly their dynamism, it is essential to find a new approach to support the evolution of these systems and to facilitate their growth and update in execution time. OVAMAH is a platform for developing open MAS. It can integrate external systems that offer services that must be completed by a particular deadline, as specified by the requesting client. These services should be analyzed to ensure that they are carried out appropriately. The RTA integrated within OVAMAH can offer the necessary mechanisms for performing a viability analysis and ensuring a correct execution, complying with the temporal restriction. OVAMAH represents the first step in obtaining true deployed virtual organizations. This study presents the next step in the evolution of OVAMAH to support SMA in execution time.

Acknowledgements. This research has been partially supported by the project TIN 2009-13839-C03.

References

1. Argente, E., Botti, V., Carrascosa, C., Giret, A., Julian, V., Rebollo, M.: An Abstract Architecture for Virtual Organizations: The THOMAS project DSIC-II/13/08 (2008)
2. Boley, H.: The rule-ml family of web rule languages. In: Alferes, J.J., Bailey, J., May, W., Schwertel, U. (eds.) PPSWR 2006. LNCS, vol. 4187, pp. 1–17. Springer, Heidelberg (2006)
3. Carrascosa, C., Giret, A., Julian, V., Rebollo, M., Argente, E., Botti, V.: Service Oriented MAS: An open architecture. In: Proc. of AAMAS 2009, pp. 1291–1292 (2009)
4. Dignum, V., Dignum, F.: A landscape of agent systems for the real world. Technical report 44-cs-2006- 061, Institute of Information and Computing Sciences, Utrecht University (2006)
5. Julián, V., Botti, V.: Developing real-time multi-agent systems. Integrated Computer-Aided Engineering 11, 135–149 (2004)
6. Lehoczky, J.P., Sacha, L., Ding, Y.: An optimal algorithm for scheduling soft-aperiodic tasks in fixed-priority preemptive systems. In: Proceedings of the IEEE Real-Time Systems Symposium, pp. 110–123 (1992)
7. Liu, C., Layland, J.W.: Scheduling algorithms for multiprogramming in a hard real-time environment. Journal of ACM 20(1), 46–61 (1973)
8. Navarro, M., Del Val, E., Rebollo, M., Julián, V.: Agent Negotiation Protocols in Time-Bounded Service Composition. In: Corchado, E., Yin, H. (eds.) IDEAL 2009. LNCS, vol. 5788, pp. 527–534. Springer, Heidelberg (2009)
9. Navarro, M., Julián, V., Soler, J., Botti, V.: jART: A real-time multi-agent platform with rt-java. In: Proc. of IWPAAMS 2004, pp. 73–82 (2004)
10. Stankovic, J.A.: Misconceptions about real-time computing. IEEE Computer 12(10), 1–10 (1988)

PathAgent: Multi-agent System for Updated Pathway Information Integration

M. Reboiro-Jato, D. Glez-Peña, R. Domínguez, G. Gómez-López,
D.G. Pisano, C. Campos, and F. Fdez-Riverola

Abstract. Nowadays, integration of biological information coming from multiple sources is a common task for the functional interpretation of experimental results during high-throughput data analysis. In concrete, metabolic pathways information is one of the most widely used genome annotation, during post-hoc analyses of differential gene expression experiments. This paper presents a multi-agent system able to retrieve and integrate metabolic functional annotations coming from multiple available databases. Taking advantage of the multi agent architecture design, the developed tool is flexible, scalable and extensible to adopt more databases. Interoperability with external systems is achieved through a freely available RESTful API.

Keywords: Multi-agent system, biological pathway, knowledge integration, functional genomics.

1 Introduction

Since the advent of the microarray technology [1] many genome-scale experiments are being carried out in order to discover the biological mechanisms hidden under

---

M. Reboiro-Jato · D. Glez-Peña · C. Campos · F. Fdez-Riverola
SING Group, Informatics Department, University of Vigo
Ed. Politécnico, Campus Universitario As Lagoas s/n, 32004, Ourense, Spain
e-mail: mrjato@uvigo.es, dgpena@uvigo.es, ccampos@uvigo.es, riverola@uvigo.es

R. Domínguez
Informatics Unit
Complexo Hospitalario de Ourense
C/ Ramón Puga 54, 32005, Ourense, Spain
e-mail: ruben.dominguez.carbajales@sergas.es

G. Gómez-López · D.G. Pisano
Bioinformatics Unit (UBio), Structural and Biocomputing Programme
Spanish National Cancer Research Centre (CNIO)
C/ Melchor Fernández Almagro 3, 28029, Madrid, Spain
e-mail: ggomez@cnio.es, dgonzalez@cnio.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 77–85.
springerlink.com

complex diseases like cancer. Differential expression experiments using microarrays usually try to compare normal vs. disease samples producing gene lists with those up- or down-regulated ones, which should be 'functionally' interpreted afterwards. Consequently, in recent years, some valuable wet-lab oriented tools have been developed in order to facilitate the functional analysis of gene lists in the biological pathways scenario. Popular examples are DAVID software [2] and FatiGO+ [3], which provide useful information and statistical assessment to understand gene lists.

In this context, integration of biological information coming from multiple sources is a common task in these kind of tools. In concrete, metabolic pathways information is one of the most widely used genome annotation, where many public databases have been created and are still growing today, such as KEGG [4], Reactome [5], NCI-PID [6], Biocarta [7], among others. In order to develop pathway-based functional analysis applications, it is necessary to integrate heterogeneous data models into a common information schema, in order to provide an homogeneous information layer, where user interfaces can be easily constructed and maintained. The most common approach is the 'data warehousing' architecture, which (*i*) defines an unified data model and (*ii*) feeds it with previously downloaded and transformed data coming from the targeted sources. However, this solution poses some drawbacks. First, data becomes outdated quickly, since original pathway sources are updated frequently and, second, it is not always easy to download and/or understand the original model. Instead of using a data-warehousing approach, a 'federated' [8, 9] architecture extracts and formats only the necessary data from different sources on-demand, often by querying a public API or by parsing HTML results. This approach has several advantages, including the fact of always retrieving up-to-date information and the absence of periodically duplicating huge databases. However, the federated architecture also poses some disadvantages, related with a poorer performance, the possibility of temporal unavailable third-party databases, and the need to update the search routines that parse/reformat results whenever a source database change their data structure or format.

In this work, we present PathAgent, a multi-agent system for functional analysis of gene lists, which integrates pathway information coming from different databases by following a federated approach. The proposed platform was implemented using the JADE [10] agent framework, taking advantage of its built-in FIPA [11] standard interaction protocols and its seamless integration of custom ontologies within agent communication acts. In addition, PathAgent HTML extraction agents are based in an in-house specific web crawler/scraping library, facilitating the creation and maintenance of the extraction routines from those databases where there is no a public access API.

The paper is organized as follows. First we describe the PathAgent architecture, detailing the different agent roles, their communication ontology and their interactions. The third section explains the obtained results, by briefly introducing the user interface. Finally, the conclusions are presented in the fourth section.

2 PathAgent Architecture

PathAgent defines several agents performing different roles in both the server and the client side. As a JADE based multi-agent system, the server side can be easily distributed among several machines, where each agent can run in a different node, even on a different network.

This section will describe the agent roles present in the platform, their communication ontology and their inter dependencies.

2.1 Agent Roles

The proposed platform makes use of typical agent roles, such as user and broker roles, whereas some others are more problem specific. Table 1 summarizes this roles, detailing the main objectives and behaviours of each one.

Table 1 Agent roles in the system PathAgent

Agent	Description
 UserAgent	Objectives: • Provide a graphical user interface to the user via a Java applet, running within the web browser. • Implement graphical representations of the relationships between the queried genes and the pathways where they are involved. • Give the user several functional analysis tools, such as statistical tests in order to assess the enrichment of the retrieved pathways. • Allow the user to export the obtained data to plain text files. Behaviours: • Uses the DF service in order to detect a BrokerAgent. • Communicates with the BrokerAgent via FIPA-Request interaction protocols.
 RESTAgent	Objectives: • Provide access to the PathAgent platform to external systems via a RESTful protocol to ensure interoperability. Behaviours: • Uses the DF service in order to detect a BrokerAgent. • Communicates with the BrokerAgent via FIPA-Request interaction protocols in order to attend REST-based requests.
 DictionaryAgent	Objectives: • Provide a gene identifier converter facility between Ensembl, Entrez, Swiss-prot, HGNC and probeset namespaces.

Behaviours:

- Registers itself in the DF.
- Receives queries via FIPA-Request in order to compute all possible names of a given gene.

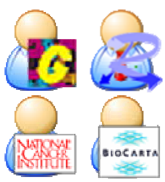
BrokerAgent

Objectives:

- Be a façade in front of all DB-specific agents. User oriented agents should use this broker agent to get pathways of multiple data sources by only one request.

Behaviours:

- Registers itself in the DF.
- Uses the DF service in order to detect all available DB-specific agents.
- Communicates with user agents, attending to their queries performed via FIPA-Request interactions.
- Communicates with the DB-specific agents via FIPA-Requests, performing simultaneous calls and gathering responses.
- Communicates with the Dictionary agent via FIPA-Request to populate gene instances with all their possible names.



DB-specific Agents

Objectives:

- Hide the underlying details of on-line access to a given pathway resource.
- Be highly configurable in order to re-adapt their access routines when the target databases change their format and/or protocol.

Behaviour:

- Registers itself in the DF.
- Communicates with the broker agent, attending to their queries performed via FIPA-Request interactions.

2.2 *Ontology*

In order to standardize the communication between agents during the FIPA protocols, we have also designed a simple ontology, using the Protégé tool [12]. The actions and concepts defined in the ontology are summarized bellow:

- Concept *PathwaySet*. Defines a set of pathways, usually retrieved after a specific query.
- Concept *GeneSet*. Defines a set of genes, usually needed to search for pathways where those genes are involved.

- Concept *Pathway*. Defines a pathway. This entity has an unique identifier, a name, the source database where it is defined and an optional URL to obtain additional information on this external resource. A pathway is linked to all involved genes.
- Concept *Gene*. Defines a gene in several namespaces. The current namespaces defining a gene are: Ensembl, Entrez, Swiss-prot, HGNC, and affymetrix probeset. A gene is linked to all pathways where it is involved.
- Action *GetPathway*. Used to request all the pathways that can be provided by a given agent.
- Action *GetPathwaysForGenes*. Used to request both the broker and the specific-DB agents to find pathways related to a given *GeneSet*. This action returns a *PathwaySet* as result.
- Action *GetGenesForPathway*. Used to request both the broker and the specific-DB agents to find genes involved in a given *Pathway*. This action returns a *GeneSet* as result.
- Action *PopulateGene*. Used to request the dictionary to give all possible names to a given *Gene*, taken as input. This action populates a *Gene* instance with identifiers in those fields with no value.

Figure 1 shows the ontology schema with concepts and actions and their relations.

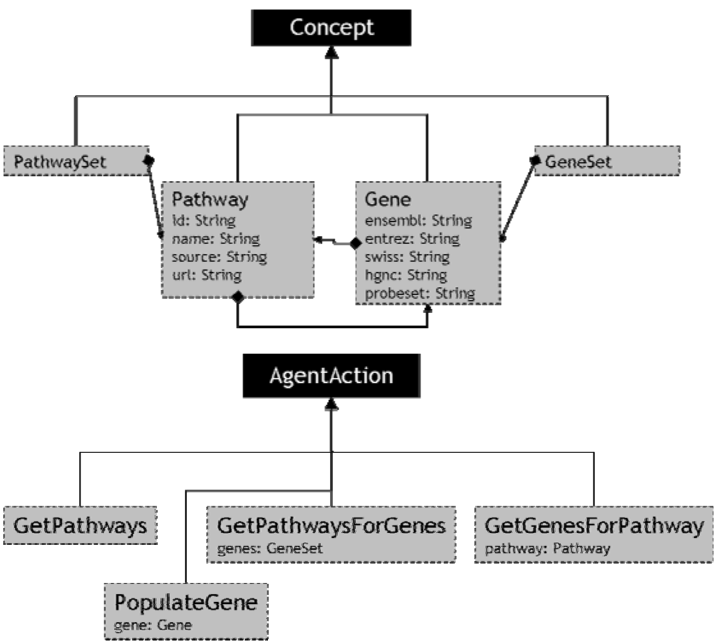


Fig. 1 High-level ontology used during the communication protocols taking place between agents

2.3 Agent Inter-Communication

The PathAgent overall architecture can be seen as a tiered client/server application, where different agents work in a specific layer. Figure 2 depicts this final architecture.

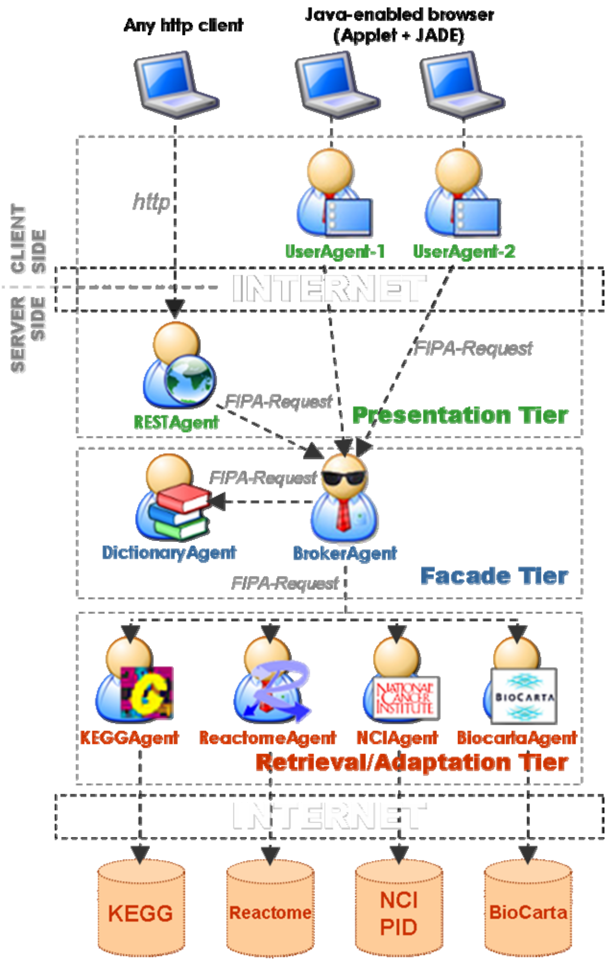


Fig. 2 Overall system architecture of PathAgent

Following a bottom-up description, the *Retrieval/Adaptation Tier* contains DB-specific agents extracting information on-demand from the external databases. For the KEGG and Reactome resources, we have made use of their publicly available APIs. For the NCI-PID and Biocarta databases, we have extracted the needed information by directly parsing the NCI website, which includes not only their own pathways, but also the Biocarta ones, thus we can extract pathways by only parsing one site. As it was previously mentioned, the HTML scraping and parsing routine is implemented via aAutomator, an in-house Java library [13].

The *Façade Tier*, offers an more amenable access service for user agents by finding all available DB-specific agents and integrating their information by (*i*) giving a unique list of pathways for a list of genes, (*ii*) retrieving all genes of a given pathway by redirecting the request to the proper DB-specific agent and (*iii*) normalize gene names by giving all their possible names.

The *Presentation Tier*, contains user agents offering two interfaces to the PathAgent users. At the client side, the UserAgent runs inside a Java applet within the user web browser. Alternatively, PathAgent can be accessed by the REST protocol through the RESTAgent, running at the server side.

3 Results and Discussion

PathAgent gives the user an intuitive and powerful graphical user interface running inside his/her Java-enabled web browser. Figure 3 contains three snapshots of the GUI.

Fig. 3 Aspect of the user interface provided by PathAgent

The main features are (*i*) interactive graphs linking genes and pathways, (*ii*) statistical functional analysis module and (*iii*) exporting facility which allows to export gene sets for further usage in, for example, gene-set enrichment tools.

4 Conclusions

This paper has presented PathAgent, a multi-agent system implementing a federated approach to the retrieval, integration of pathway information plus an interactive graphical environment. By taking advantage of the MAS paradigm and the JADE development framework, we could easily design and deploy a highly distributed and scalable system, which also integrates a communication ontology.

With this environment, biologists has a new freely accessible tool to functionally interpret gene lists by using always updated annotation information, that can also be easily extended to more pathway databases with the addition of more DB-specific agents. Further work is focused on (*i*) adding more pathway sources (*ii*) extend the overall architecture and ontology to include more annotations, like GO, disease knowledge, etc. and (*iii*) improve the user interface by performing more statistical analysis techniques.

Acknowledgements. This work is supported in part by the project Development of biomedical applications from University of Vigo (09VIB10). D. Glez-Peña acknowledges Xunta de Galicia (Spain) for the program María Barbeito.

References

1. Quackenbush, J.: Microarray analysis and tumor classification. The New England Journal of Medicine 354, 2463–2472 (2006)
2. Dennis Jr., G., Sherman, B.T., Hosack, D.A., Yang, J., Gao, W., Lane, H.C., Lempicki, R.A.: DAVID: Database for Annotation, Visualization, and Integrated Discovery. Genome Biology 4(5), 3 (2003)
3. Al-Shahrour, F., Minguez, P., Tárraga, J., Medina, I., Alloza, E., Montaner, D., Dopazo, J.: FatiGO +: a functional profiling tool for genomic data. Integration of functional annotation, regulatory motifs and interaction data with microarray experiments. Nucleic Acids Res. 35(Web Server issue), W91–W96 (2007)
4. Kanehisa, M., et al.: KEGG for linking genomes to life and the environment. Nucleic Acids Research 36(Database issue), D480–D484 (2008)
5. Matthews, L., et al.: Reactome knowledgebase of human biological pathways and processes. Nucleic Acids Research 37(Database issue), D619–D622 (2009)
6. Schaefer, C.F., et al.: PID: the pathway interaction database. Nucleic Acids Research 37(Database issue), D674–D679 (2009)
7. Biocarta: Charting Pathways of Life, `http://www.biocarta.com`
8. Zhang, Z., Cheung, K., Townsend, J.: Bringing Web 2.0 to bioinformatics. Briefings on Bioinformatics 10(1), 1–10 (2008)
9. Glez-Peña, D., et al.: WhichGenes: a web-based tool for gathering, building, storing and exporting gene sets with application in gene set enrichment analysis. Nucleic Acids Res. 37(Web Server Issue), W329–W334 (2009)

10. JADE: Java Agent Development Framework, `http://jade.tilab.com/`
11. FIPA: Foundation for Intelligent Physical Agents, `http://www.fipa.org`
12. `http://protege.stanford.edu/`
13. Glez-Peña, D., Méndez, J.R., Fdez-Riverola, F.: aAUTOMATOR: Herramienta flexible para la extracción de información en sitios web bioinformáticos. In: Conferencia Ibero-Americana WWW/Internet 2007: CIAWI 2007 (2007)

Multi-agent System for Mass Spectrometry Analysis

Miguel Reboiro-Jato, Daniel Glez-Peña, Hugo M. Santos, Mário S. Diniz, Carlos Lodeiro, José L. Capelo, and Florentino Fdez-Riverola

Abstract. This paper presents a framework for rapid distributed application development in mass spectrometry data analysis. The proposed system includes an extensible and flexible workflow that covers the whole data analysis process. This workflow was designed to be easily distributed and parallelized by implementing the underlying framework as a multi-agent system.

Keywords: mass spectrometry, framework, data analysis workflow, multi-agent system.

1 Introduction

In recent years, proteome analysis has become a very important tool for identifying and quantifying biomarkers that may ease diagnosis and monitoring of various disease states [1]. Among all the proteome analysis techniques, mass spectrometry (MS) in combination with liquid chromatography (LC) stands out for being the

---

Miguel Reboiro-Jato · Daniel Glez-Peña · Florentino Fdez-Riverola
SING Group, Informatics Department, University of Vigo
Ed. Politécnico, Campus Universitario As Lagoas s/n, 32004, Ourense, Spain
e-mail: mrjato@uvigo.es, gpena@uvigo.es, iverola@uvigo.es

Hugo M. Santos · Mário S. Diniz · Carlos Lodeiro · José L. Capelo
REQUIMTE, Departamento de Química, Faculdade de Ciências e Tecnologia, FCT, Universidade Nova de Lisboa, 2829-516. Caparica, Portugal
e-mail: hmsantos@dq.fct.unl.pt, mesd@fct.unl.pt

Carlos Lodeiro
BIOSCOPE Group, Physical Chemistry Department, Science Faculty, University of Vigo, Ourense, Spain
e-mail: clodeiro@uvigo.es

José L. Capelo
BIOSCOPE Group, Nutrition & Bromatology Area, Analytical & Food Chemistry Department, Science Faculty, University of Vigo at Ourense Campus, Spain
e-mail: jlcapelom@uvigo.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 87–95.
springerlink.com

key analytical technique on which several emerging technologies, such as proteomics, metabolomics and lipidomics, are based [2]. Due to the continuous enhancement of the assay instruments both the quality of the results and the amount of data generated have increased greatly. In order to manage all these data, bioinformatics applications have become an essential tool for researchers in the proteomic field.

In this context, the steps needed on each analysis and their parameters are not deterministic even if the same assay technique or instrument is used. Researchers are responsible for establishing which phases are applied and their parameters depending on the characteristics of the samples they want to study. Defining the steps to follow and correctly applying them are not trivial tasks and sometimes become a ‘trial and error’ process being a time-consuming phase of the whole experiment process.

Although over the last decade several successful bioinformatics tools for proteomics data analysis have been developed [3], many of them are mainly focused on specific data analyses and they do not cover the whole process nor are flexible enough to be adapted to new experiments [4]. These limitations imply that researches usually have to use several analysis tools together with conversion utilities in order to share data between applications, therefore increasing the time employed on the analysis phase.

Another problem evidenced in commercial applications is the use of both proprietary and/or private data formats and algorithms, which leads to difficulties in sharing data with other tools and also that the researcher has not the complete knowledge about the processes being applied. In contrast, open source tools use public data formats and algorithms but with certain exceptions [4, 5], most of them are focused on one step of the whole analysis process.

In order to speed up the analysis process and to correctly support the increasing amount of data generated by the analysis tools, a good number of proposals have recently emerged for building distributed systems for data analysis in the field of mass spectrometry [7, 8]. The first studies demonstrated the validity of these approaches and the improvement on the overall performance achieved with their utilization [9].

Encouraged by these results, in this paper we present MaSAS (*Mass Spectrometry Analysis System*) a framework designed for rapid distributed application development in proteomics data analysis. MaSAS provides the entire underlying infrastructure needed by developers in order to avoid the codification of new methods for data processing and analyzing from scratch. Moreover, MaSAS also facilitates both the definition of the sequence steps to be applied in an experiment and the creation of new graphical interfaces to visualize the analysis results. The proposed framework presents several similarities with the OpenMS system [4] but MaSAS is designed to run either in a standalone mode or as a distributed system also providing the researcher with a graphical interface to set-up and configure all the analysis steps.

The rest of the paper is structured as follows: Section 2 presents the background surrounding the system design. Section 3 explains the architecture of the developed multi-agent system. Finally, Section 4 summarizes the conclusions and establishes future research lines.

2 System Design

The design of our framework was mainly focused on the definition of a highly parallelizable and distributable workflow. The main goal was the creation of a system easy to use by researchers and easy to extend by developers, being able to grow inside a research group with minimum help from developers.

MaSAS is structured in five parts, namely: (*i*) a data model, (*ii*) a workflow that defines how the steps are executed, (*iii*) a set of interfaces for the implementation of workflow's steps, (*iv*) a web interface, and (*v*) a set of utilities for workflow managing. In the next subsections we will explain in detail the system architecture and how our design principles where correctly achieved.

2.1 Mass Spectrometry Sample Study

In this subsection we are going to briefly describe the type of mass spectrometry studies for which our framework was designed. A typical mass spectrometry case study may be divided in four main phases, namely: measurement, spectra, sample and experiment.

Measurement phase
A mass spectrometer measures the mass-to-charge ratio (*m/z*) of gas-phase ions. In a mass spectrometry experiment, the measurement of the abundance against *m/z* (peak) for each ion generates a 'mass spectrum' (spectrum) [2]. The abundance is usually measured as an intensity, which may be absolute or, more commonly, relative. In order to improve the reliability of the obtained data, several replicates of the analysis may be done. So, for each sample, a set of spectra is generated in this step of the analysis.

Spectra phase
Once the spectra are generated for a given sample, the data is processed in order to remove noise that may contaminate the analysis. If several replicates where done on the previous phase, a 'peak alignment' step is applied in order to remove false-positive peaks. In general, false-positive peaks do not repeat themselves well across multiple spectra. Every time all the identified peaks coming from multiple spectra (including false-positive ones) are aligned, false-positive peaks are detected and removed because they are not as consistent as true peaks [10]. The result of the alignment is a single representative spectrum of the whole sample (sample).

Sample phase
The main goal of this phase is to individually analyze the generated samples. Each sample is treated as independent. Sometimes the data study ends with this step, for example, when the objective is to identify those peptides present in a sample.

Experiment phase
Every time the experiment requires the analysis of data coming from several samples (bi-class or multiple-class studies), a phase to compare existing samples is

required. There are many techniques that can be applied in this phase (i.e. heat map generation, clustering algorithms, etc.) depending of the experiment type.

2.2 Data Model

The MaSAS data model reflects the evolution produced in the mass spectrometry data generated by the analysis process described above. The classes of the model are (*i*) *Measurement*, which represents a peak measured by the mass spectrometer, (*ii*) *Spectrum*, which stands for a set of peaks measured in the analysis of a single sample, (*iii*) *Spectra*, which contains a set of Spectrum and represents several replicates of the analysis made to a single sample, (*iv*) *RepresentativeMeasurement*, which is a special kind of *Measurement* used as representative for a set of aligned peaks, (*v*) *Sample*, which contains a set of *RepresentativeMeasurement* symbolizing the characteristic spectrum of a single sample, and (*vi*) *Experiment*, composed by a set of *Samples* each one corresponding to an analyzed sample. Spectra, Sample and Experiment types have a special importance due to the fact that, as we will see, they define the three main stages of the workflow.

The proposed data model is complemented with a set of classes for collecting the results of the analyses. Because of the variety of the analyses that can be applied (i.e. statistical analysis, chart plotting, etc.) the type of the results generated is also very variable. *AnalysisResult* objects act as a container with a single property called 'result' and a reference to the element being analyzed.

There are three specific result classes: (*i*) *SpectraAnalysisResult* generated by the Spectra analyses, (*ii*) *SampleAnalysisResult* generated by the Sample analyses, and (*iii*) *ExperimentAnalysisResult* generated by the Experiment analyses.

2.3 Step Interfaces

MaSAS framework defines three different kinds of data manipulation operations: (*i*) processes, (*ii*) combinations, and (*iii*) analyses. Processes are procedures that perform some action over a given input data object and return a new data object of the same type. Combinations are useful for processing an input data object and returning a new data object of a different type. Analyses are intended for processing an input data object and generating an output AnalysisResult object (without modifying the input object).

2.4 MaSAS Workflow

The workflow architecture is based on the mass spectrometry case study described above. Divided into several atomic steps, the workflow starts with a first data loading step, followed by three main stages: Spectra, Sample and Experiment. The transition between these stages is done by two combination steps. Fig.1 shows a graphical representation of the workflow, where dark green boxes represent the stages and light green boxes represent the steps. As it can be seen, each stage is composed of both process and analysis steps.

The workflow was divided into several atomic steps in order to both reduce its complexity and allow parallel and separated execution of the steps. Atomic step components are also easy to implement by developers (and even by a researcher with programming skills), as they only have to implement simple interfaces. This architecture also enables the custom ordering and configuration of steps inside stages. The steps and stages of the workflow are the following:

- *Load Data Step*: mass spectrometry data is loaded. Generated mass spectra must be provided for each sample. The loaded data is initially represented with a Spectra instance.
- *Spectra Analysis Stage*: this stage is divided into two different steps: (*i*) *processing* where SpectraProcess components are executed, and (*ii*) *analyzing* where SpectraAnalysis components are executed. Any number and order of steps are allowed.
- *Spectra Combination Step*: Spectra objects become Sample objects in this step by executing a SpectraCombination component. Spectra combination usually implies the execution of a peak alignment algorithm. The result of this step is a Sample instance representative of the sample.
- *Sample Stage*: this stage is quite similar to the Spectra Stage, with the only difference that working data is now Sample data. This implies that in processing steps SampleProcess components are executed and, in analyzing steps SampleAnalysis components are executed.
- *Sample Combination*: sample instances coming from the different samples analyzed in the experiment are combined by a SampleCombination component that generates a new Experiment instance.
- *Experiment Stage*: multiple sample analyses and processing are done in this stage. ExperimentProcess and ExperimentAnalysis components are executed in the process and analysis steps, respectively.

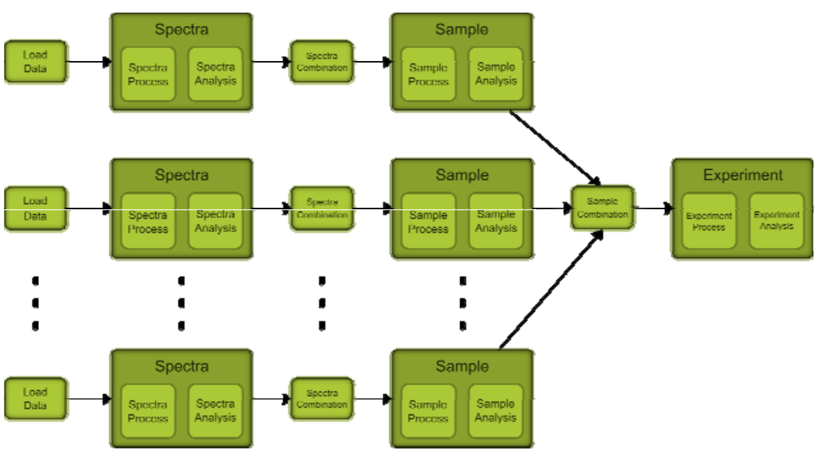


Fig. 1 MaSAS workflow

2.5 Web Interface

MaSAS includes a web interface where researchers can (*i*) load and analyze data step by step, (*ii*) design a new workflow and store it for further executions, (*iii*) load sample data and automatically apply a stored workflow, and (*iv*) view the results generated in the analysis.

The web interface was designed to be easily extended with new processes, analyses, combinations and result viewers. Developers can incorporate new components to the web just by adding a few lines to a plain configuration file, making it available to researchers.

The web interface was implemented using ZK framework [11], a web framework based in Java, JavaScript and AJAX. Result viewers can be created by employing ZK resources like the ZUML language or a Java user interface API.

3 Multi-agent System Architecture

One of the main design goals with MaSAS was to be able to operate as a distributed system. This section explains in detail the four layers which constitute the MaSAS multi-agent architecture and also identifies different types of agents. Fig. 2 gives a general overview about the proposed architecture.

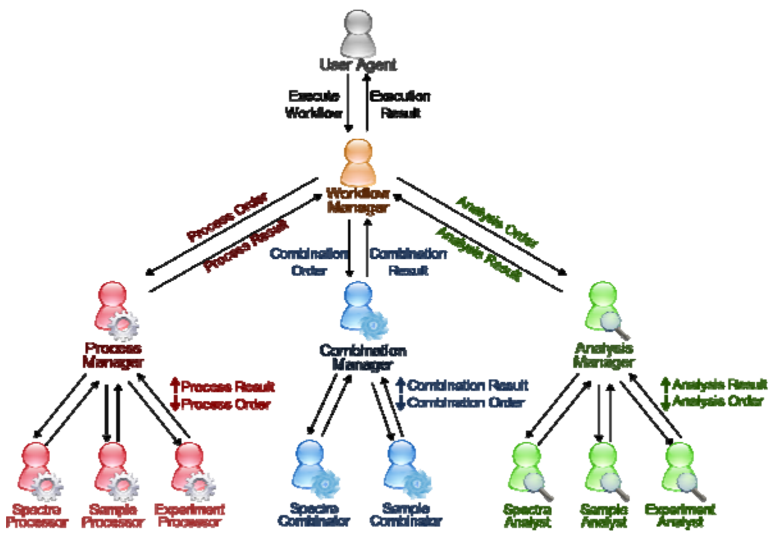


Fig. 2 Multi-agent system architecture

First layer contains User Agents. These are interface agents with three main responsibilities: (*i*) allow the user to design new workflows, (ii) allow the user to load data and apply a workflow, and (*iii*) show the result data generated in a workflow execution. MaSAS provides a User Agent implemented as a Web agent based in the ZK framework, as commented above.

Internal workflow execution is the responsibility of bottom layers. Two intermediate layers distribute the work load, while a last layer contains agents that actually perform the work. First intermediate layer contains the Workflow Manager Agents. Its main function is to receive workflows coming from User Agents and distribute the work between agents of the next layer, returning as result the data generated. Workflow Manager Agents also provide User Agents with information about the available Execution Agents, needed by researchers when building workflows.

Second intermediate layer contains the Execution Manager Agents, divided into Process Managers, Combination Managers and Analysis Managers, according to the type of data that they manage. These agents receive single executable steps and distribute them among the appropriate Execution Agents of the next layer. In the same way that Workflow Manager Agents provide User Agents with information about available Execution Agents, Execution Manager Agents provide Workflow Manager Agents with information about their managed Execution Agents.

The last layer contains the Execution Agents that perform the data processing, combining and analyzing algorithms. These agents implement the interfaces presented in subsection 2.3, which determines their type. A single agent can implement several steps and/or code several algorithms for the same step, so it can act as different Execution Agent types. By replicating or distributing these agents between different machines, the work load can be easily distributed.

Within this context, a typical working scenario is divided into two phases: (*i*) experiment design and (*ii*) experiment execution. Experiment design is done by the user through the User Agent. In this phase the user needs to know which algorithms are available for each step. This information is queried to the Workflow Manager Agent that, in turn, queries the Execution Managers for available Execution Agent types and algorithms. Once the experiment is designed, the user can load data and execute the associated workflow. Workflow execution is directed by a Workflow Manager, which distributes (and parallelizes) the step execution through the underlying Execution Manager Agents. At the same time, Execution Manager Agents distribute the steps through the Execution Agents. Every time a step is executed, the generated results reach the User Agent following the reverse path.

Agent discovery is done by using Directory Facilitator agents, both for the location of Execution Manager Agents and Execution Agents. Execution Manager Agents must register themselves indicating the data type they are able to manage, while Execution Agents must register the type of the step they implement.

The designed architecture allows researchers and companies to publish their own agents implementing their algorithms for a particular step of the workflow, while other researchers can use these agents in order to perform a data study, integrating them into their self-designed workflows. Private companies can also take advantage of this architecture by controlling the access to their agents (i.e.: granting access only to those who bought a concrete hardware).

4 Conclusions and Further Work

This work presents the main issues in the design of the MaSAS multi-agent framework, a distributed system for mass spectrometry analysis. MaSAS incorporates an innovative workflow that can be easily extended with new algorithms and processes, but also allows researchers to create their own experiments.

The proposed architecture enables the distribution and parallelization of the workflow execution. Moreover, it is able to address the dynamical incorporation of new agents by implementing some of the workflow steps or managing the execution.

Future work will aim towards the expansion of the proposed workflow in order to cover new mass spectrometry study cases. Another future research line is the creation of a core library of agents implementing the most common algorithms for processing, combining and analyzing mass spectrometry data.

Acknowledgements. This work is supported in part by the project Development of biomedical applications from University of Vigo (09VIB10) and Xunta de Galicia-Conselleria de Economía e Industria (Spain) by project 09CSA043383PR. H. M. Santos acknowledges the doctoral grant SRFH/BD/38509/2007 provided by FCT (Science and Technological Foundation) of Portugal. D. Glez-Peña acknowledges the Xunta de Galicia (Spain) for the program María Barbeito. Dr. José-Luis Capelo-Martínez and Dr. Carlos Lodeiro are grateful to the Xunta de Galicia (Spain) for the program Isidro Parga Pondal and the University of Vigo for financial support under projects InOu-Univ.Vigo 2009-K915 and K914. M. Diniz acknowledges financial support given by REQUIMTE (Portugal).

References

1. Findeisen, P., Neumaier, M.: Mass spectrometry based proteomics profiling as diagnostic tool in oncology: current status and future perspective. Clinical Chemistry and Laboratory Medicine (2009)
2. Griffiths, W.J., Wang, Y.: Mass spectrometry: from proteomics to metabolomics and lipidomics. Chemical Society reviews (2008)
3. Xu, C., Ma, B.: Software for computational peptide identification from MS-MS data. Drug Discovery Today (2006)
4. Sturm, M., Bertsch, A., Gröpl, C., Hildebrandt, A., Hussong, R., Lange, E., Pfeifer, N., Schulz-Trieglaff, O., Zerck, A., Reinert, K., Kohlbacher, O.: Open MS – An open-source software framework for mass spectrometry. BMC Bioinformatics (2008)
5. Kohlbacher, O., Reinert, K., Gröpl, C., Lange, E., Pfeifer, N., Schulz-Trieglaff, O., Sturm, M.: TOPP – the OpenMS proteomics pipeline. Bioinformatics (2006)
6. Veltri, P., Cannataro, M., Tradigo, G.: Sharing mass spectrometry data in a grid-based distributed proteomics laboratory. Information Processing & Management (2007)
7. Cannataro, M., Veltri, P.: A Grid Service for Pattern Extraction from Mass Spectrometry Data. In: Second International Conference on Semantics, Knowledge, and Grid (2006)

8. Quandt, A., Masselot, A., Hernandez, P., Hernandez, C., Maffioletti, S., Appel, R.D., Lisacek, F.: SwissPIT: An workflow-based platform for analyzing tandem-MS spectra using the Grid. Proteomics (2009)
9. Quandt, A., Hernandez, P., Kunzst, P., Pautasso, C., Tuloup, M., Hernandez, C., Appel, R.D.: Grid-based analysis of tandem mass spectrometry data in clinical proteomics. Studies in health technology and informatics (2007)
10. Yu, W., He, Z., Liu, J., Zhao, H.: Improving Mass Spectormetry Peak Detection Using Multiple Peak Alignment Results. Journal of Proteome Research (2007)
11. ZK – Direct RIA, `http://www.zkoss.org` (Last visit: 18/11/2009)

Multi-agent Personal Memory Assistant

Ângelo Costa, Paulo Novais, Ricardo Costa, Juan M. Corchado, and José Neves

Abstract. Memory is one of our most precious goods has it gives us the ability to store, retain and recall information thus giving a meaning to our past and help us to envision our future, dreams and expectations. However, ageing decreases the capacity of remembering and the capacity to store new memories, thus affecting our life quality. These presented problems configure a social and human dilemma. With the presented work we intend to address some of these problems, thru the use of the Personal Memory Assistant (PMA) concept in order to help its user to remember things and occurrences in a proactive manner. We will also address socialization and relaxation events that should be part of the user's life. With the use of a Multi-Agent System to implement the PMA, the objectives can be achieved in a ubiquitous and highly configurable manner. It is presented here the platform concept, scheme and the agent characteristics and their contribution to each and every agent.

Keywords: Multi-agent systems, e-Health, Memory Assistant, Ambient Assisted Living, Scheduling.

1 Introduction

An intrinsic characteristic of the human being is his memory. Our remembering capacity has the utmost importance as it gives us the sense of being and the capacity to have a social life. It makes objects to have a meaning and it keeps the

---

Ângelo Costa · Paulo Novais · José Neves
CCTC, Departamento de Informática, Universidade do Minho, Braga, Portugal
e-mail: angelogoncalocosta@gmail.com, {pjon,jneves}@di.uminho.pt

Ricardo Costa
CIICESI, College of Management and Technology of Polytechnic of Porto, Felgueiras, Portugal
e-mail: rcosta@estgf.ipp.pt

Juan M. Corchado
Departamento Informática y Automática, University of Salamanca, Salamanca, Spain
e-mail: corchado@usal.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 97–104.
springerlink.com

instructions of how things are done. This work is oriented to an older population, typically retired, helping them in scheduling events and suggest activities, in order to fill their unfilled time, with minimal interaction. To achieve this we have created a platform based on a Multi-Agent System that can, through embedded intelligent and recurring to other collaborative agents, address it. On top of this platform a Personal Memory Assistant and a Social Enabler where developed and are here presented. We are also going to demonstrates that a distributed system approach is adequate for developing multi-agent systems for healthcare provision tools, helping improving the life of its users, focusing on the PMA paradigm.

2 Ageing Factors

The life expectancy of the world population, according to the United Nations Population Fund (UNPF), is increasing and the birth rate of children is decreasing rapidly. The UNPF estimates that in a period of fifty years the European Population decreased 13%, increasing the age average to 48 years old [1]. At the age of 50 the human beings begin to be affected by it, being the forgetfulness of more re-cent events, one of the most occurred symptoms.

In most cases the loss of memory is more likely to appear and will keep advancing in parallel with ageing and there is little to do in terms of regaining the memorizing capabilities, the most it can be done is exercising the brain through specific brain training exercises [2]. There are three main stages of loss of memory: Cognitively Unimpaired, Mild Cognitive Impairment and Severe Cognitive Impairment (Dementia). Each stage requires different attention and focus suited to their conditionings. Additionally we should also consider the possible memory problems associated with diseases, especially chronic ones, that affects our ability to remember, meaning we can have younger people affected with this problem.

There is still no known way of reversing the human brain loss of information, so a possible solution may be the use of computational systems to store and retrieve all that data. As studies have been done in the area of human interaction and wellbeing, it is know that scheduling and storing, intelligently, user's activities makes communication easier with their peers and relatives, thus greatly improve the elderly self-esteem on their daily activities [3] [4].

3 iGenda: Scheduling and Organization

The main objective in this work is to present an intelligent scheduler that inter-acts with its users through computational means, creating a product that will help remember relevant information and events, a PMA [5]. This is done by trying to emulating the way the brain processes new events and reorganizes already scheduled ones. To reschedule an event we normally see in the agenda which are the events able to be moved to another place, considering factors like their relevance and possible problems (e.g. other persons involved in a specific event). It is specially aimed to help the ones with loss of memory, by sustaining all the daily events and warning the user when it is time to put them into action. It will be able to receive information delivered by any source and organise it in the most convenient way, according to predefined standards and protocols, so that the user will not need to manually plan or schedule specific events and tasks. The iGenda is a hierarchy of Agents (Fig. 1) that follow specific tasks and protocols [6], intended to deal with its user's expectations. The main agents are the Agenda Manager (AM), the Free Time Manager (FTM), the Conflicts Manager (CM) and the Interface Manager (IM). All the project agents were constructed to function on the JADE platform, having the decision making processes been written in Logic Programming Language (LPL), namely Prolog in order to the results to be assertive and logically correct.

Fig. 1 Modules Scheme of iGenda

We approached the problem by constructing a Multi-Agent System [6] that is able to support Communication and Decision Making Agents, each one with unique characteristics in a non-competitive and parallel functioning environment so agents can separately process different calendars and events, making different choices. They obviously cannot write to the same calendar at the same time so they are also non-cancelling each other. Also, with the utilization of JADE, it is possible to distribute agent platforms across several machines, which may have different operative systems, with the possibility to migrate them during run-time. It also provides portability, which means that any module may run in different machines, being positioned on any part of the world.

4 Agents Communication

The proposed environment is a distributed one, so it is essential to provide the necessary mechanisms for cooperation and coordination in order to allow the agents to proceed, correctly, with their tasks. The communication between the several existing agents can be established by several means of communication like Ethernet, WIFI, GSM, UMTS and others (Fig. 2).

The communication protocol complies with the FIPA (Foundation for Intelligent Physical Agents) standard ACL, being the JADE platform used to build the iGenda architecture (Fig. 3). All the agents are compliant with this standard since they are all JADE implemented agents. Messages carry the information of updates and direct announcements. These messages are sent and received through the several modules and clients. There are already some agents being developed in

JADE-Leap [7] for the needed interaction with mobile devices and also an early implementation and structuring of the project in JADEX [8] as it is our belief that the BDI will help to design and implement future agents that will be added to the system.

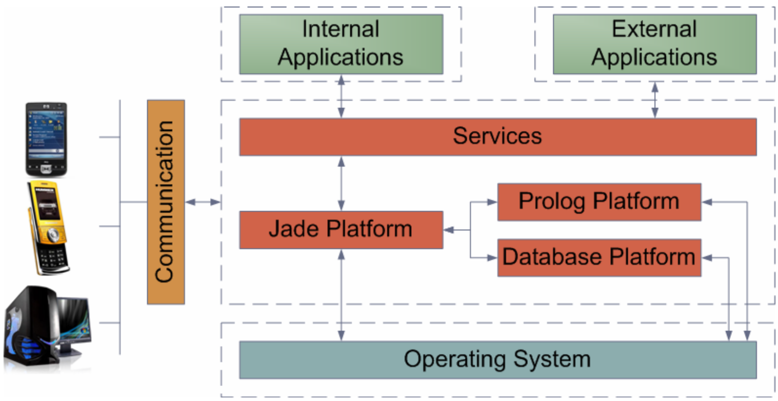


Fig. 2 Scheme of the Stages

Due to its small size the messages are lightweight and easy to be transmitted between agents using the JADE platform message buffer, where all messages are saved even if the receiver agent is not active. A table and system of capturing and interpreting errors was created in order to the errors occurring during the processing phase can be relayed.

5 Agenda Manager

The Agenda Manager (AM) sets the bridge between the remaining parts of the manager system and the scheduling one, using the communication infrastructure available to receive and send requests. As result the AM stands for the communication and security of the whole systems. It configures a two stage application agent: it manages the incoming events to be scheduled and programs the time that triggers the Free Time Manager and it is also the communication input and output to the Client agents.

The AM supports the reception of multiple messages, and when it doesn't constitute a conflict in term of accessing the same calendar, the execution of the remaining agents at the same time, thus increasing the overall performance of the system. The AM ensures also that the user's friends and relatives can keep in touch with him and know what activities he is or will be doing.

Indeed, the AM manages the entire project. Its assessment modifies the way the project works.

6 Conflicts Manager

The Conflicts Manager (CM) agent is intended to assure that there is no overlap of activities and events. This module schedules or reorganizes the events that are relayed from the Agenda Manager, insuring that they are in accordance with the rest of the events. When a collision of different events is detected, the outcome will be decided by methods of intelligent conflict management. In case of overlapping events with the same priority level, the notification of overlapping is reported to the sender, so he/she may try to reschedule to a different time slot.

The events follow a hierarchy system. Every event has a value that is defining of his priority or urgency. Most of the conflicts will use the priorities value to be solved. This agent has also the capacity to manage all the connections with the other users as well as with the user relatives.

The Conflicts Manager operation can be explained in the following way:

1. When an operation is done by an administrator, the AM receives the message and calls the CM.
2. The CM enters in action by reading all the calendar files, parsing the new event and using the CLP engine to compare the priority levels of eventually conflicting events.
3. It is created a new ICS calendar ready to be delivered to the user.
4. A new message is then sent to the user and to the administrator, notifying them that a new Calendar is available and that the new insertion was successful.

7 Free Time Manager

The Free Time Manager (FTM) will schedule recreational activities in the free spaces of the user's calendar, in order to enforce the well being of the user. These activities configure an important milestone for an active ageing on the part of the user, once it promotes cultural, educational and conviviality conducts, based on an individualized plan. The FTM has a database that contains information of the user's favourite activities, previously verified by the decision support group or a medical committee. This agent supports the reception of a time trigger in order to be activated, always sent by the AM. It also as a built-in database connection in order to be able retrieve the available events and the logic engine responsible for all the decisions and changes made to the calendar.

The FTM uses a distribution function (2) defined attending some variables allowing it to decide the activity that is inserted into the user free time. For instance, in a three activities packet, the rate for the activity, from higher to lower priority, is 70%, 25% and 5%. The activities are merely suggestive, it comes to the user to decide to execute them or not. On the other hand the activities chosen are those that fit into the time space that is available.

As the project evolved, especially during the investigation phase, we found other related projects oriented to elderly people like TimeBank [9] and ePal.

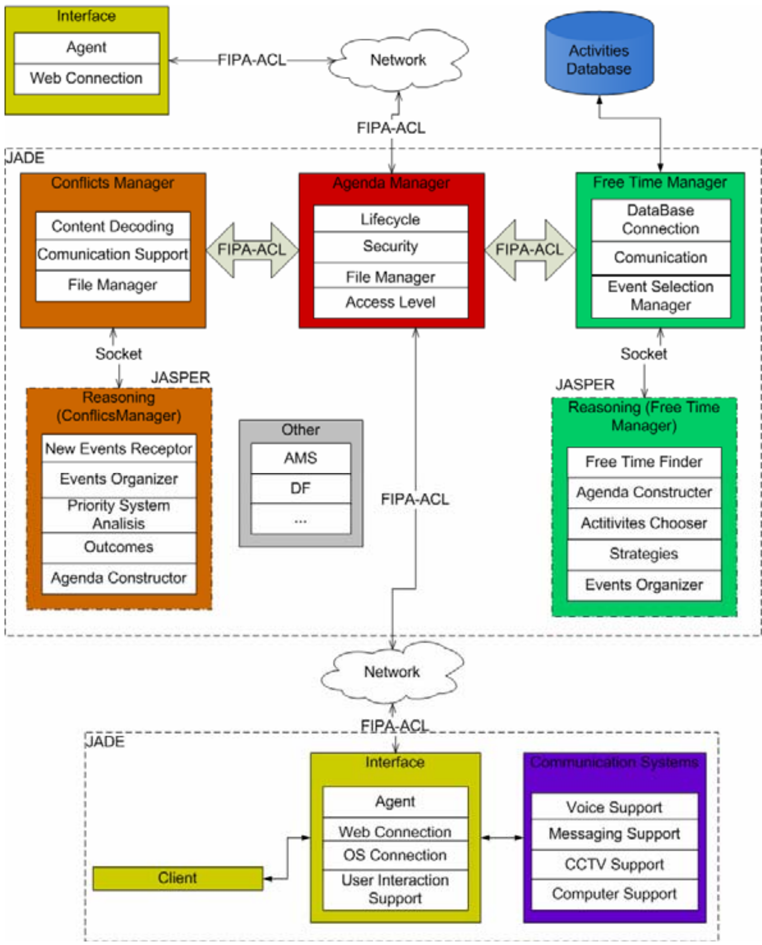


Fig. 3 Scheme of the Structure

Although these projects are not directly associated with the PMA area, their objective of keeping an interesting life through activities seemed to be a rather interesting way to the elderly. We believe these projects can be used by the FTM as activities supplier.

8 Interface Manager

The interface intends to be intuitive and easy to use. It is known that the elderly have some difficulties with new technologies, so the interface must be intuitive and easy to use. Large buttons are used and only the necessary information is displayed. A variable warning system is also available. When an event is triggered or accomplished, the user is informed.

This agent assures that the information reaches the user, by computer or mobile phone, so the user is always connected to the system. The information warnings and content can be textual, in audio format (pre-recorded messages) or both.

In terms of functionalities it supports the communication between the iGenda and supports also the distribution of the information through the already mentioned devices.

9 Conclusions

Currently most of the project bare bones are already working and we are currently focused in the remaining functionalities. Regardless of how it will evolve in the future, there are still problems and critical decisions to be made, namely the "density" problem, where by density we mean overcrowding the calendar of the user with too many activities, making it more stressful than relaxing. It also makes the difference to other PMAs, once it introduces the component of free time occupation, a problem to be addressed in terms of socialization; i.e., in terms of a process by which the user learn acceptable and unacceptable behaviours for a give environment. Also a Case Based Reasoning (CBR) system [10] [11] is under assessment to introduce the capability to record all of the user choices and actions to improve the future recommendations and actions taken as also to keep track as a log for future components improvement.

The mobile phone functionality is still under development but should be fully functional in the near future.

References

1. UNFPA: Population Ageing and Development: Operational Challenges In Developing Countries (2002)
2. Tucker, G.: Age-Associated Memory Loss: Prevalence and Implications. Journal Watch Psychiatry (1995)
3. Aguilar, J.-M., Cantos, J., Expósito, G., Gómez, P.: Tele-assistance Services to Improve the Quality of Life for Elderly Patients and their Relatives: The Tele-CARE Approach. The Journal on Information Technology in Healthcare (2004)
4. Brown, S.: Next generation telecare and its role in primary and community care. Health & Social Care in the Community 11, 459–462 (2003)
5. Costa, Â., Novais, P., Costa, R., Neves, J.: Memory Assistant In Everyday Living. In: ESM 2009 - The 23rd annual European Simulation and Modelling Conference (2009)
6. Fraile, J., Tapia, D., Rodríguez, S., Corchado, J.: Agents in Home Care: A Case Study. In: Corchado, E., Wu, X., Oja, E., Herrero, Á., Baruque, B. (eds.) HAIS 2009. LNCS, vol. 5572, pp. 1–8. Springer, Heidelberg (2009)
7. Luck, M., Ashri, R., d'Inverno, M.: Agent-Based Software Development. Artech House Publishers (2004)
8. Moreno, A., Valls, A., Viejo, A.: Using JADE-LEAP to implement agents in mobile devices, `http://jade.tilab.com/papers/EXP/02Moreno.pdf`

9. Pokahr, A., Braubach, L.: From a Research to an Industrial-Strength Agent Platform: Jadex V2. 9. Internationale Tagung Wirtschaftsinformatik (2009)
10. Castolo, O., Ferrada, F., Camarinha-Matos, L.: TeleCARE Time Bank: A Virtual Community for Elderly Care Supported by Mobile Agents. The Journal on Information Technology in Healthcare, 119–133 (2004)
11. Corchado, J., Bajo, J., Abraham, A.: GERAmI: Improving the delivery of health care. In: IEEE Intelligent Systems. Special Issue on Ambient Intelligence, 2nd edn., vol. 23, pp. 19–25 (2008)
12. Corchado, J., Bajo, J., Paz, Y., Tapia, D.: Intelligent Environment for Monitoring Alzheimer Patients, Agent Technology for Health Care. In: Decision Support Systems, 2nd edn., vol. 44, pp. 382–396 (2008)

TENSSION: A Tool for the Medical Patient Follow-Up

Del Rey Ignacio, Ana Belén Gil, and R. González-Celador

Abstract. This paper presents the tool, called "TENSSION". This tool allows an exhaustive follow-up of a patient's stay in the centre to be made, storing each of the admissions, the Services through which he or she passed and a list of daily annotations containing the most important data, to carry out statistical studies with the data from the centre in a more general way and thus have information about patient functioning and evolution and, in turn, have a means to generate alerts when possible complications arise.

Keywords: Health tools, preventive medicine, monitoring patient, medical data.

1 Introduction

Medical tools involving heterogeneous information in databases are characterized by a high degree of complexity in terms of quantity of items, number of parameter values and data types (free text, categorical, numerical and other). The Project aims to cover the demand established by the Preventive Medicine Service of the University Hospital in Salamanca. This demand responds to the need for recording all the patient information in an electronic format to provide a user-friendly patient follow-up. Here we describe a computerized patient information system developed. Before the project described in this paper the way to work was manually across different and disconnected files mainly handwritten. This process imply problems such a lack of physical space for data storage and high risk of document loss or deterioration. Sometimes, the fact that data can be kept suitably in a form amenable to being read by a machine is a type of security hard to

Del Rey Ignacio · Ana Belén Gil
University of Salamanca, Department of Computer Science and Automation – Faculty of Sciences, Plaza de la Merced s/n, 37008, Spain
e-mail: nreytb@yahoo.es, abg@usal.es

R. González-Celador
University of Salamanca, Department of Preventive Medicine and Public Health
e-mail: celador@usal.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 105–112.
springerlink.com

achieve in an environment without computers. Intensify with the absence of staff able to correctly gather the necessary data. Due to this situation exists an inability to work with the data collected and perform studies on them; and the lack of classification and ordering of the data at patient level so that patient follow-up can be accomplished from a very clear and simple distribution.

2 Motivations and Related Work

The advantages of computerized patient records have been discussed extensively in several work approaches as related in ([1], [2], [3]), with different information about experiences about the process for implementing these information systems ([4], [5], [6]). There exist diverse types of computer applications, but none responds exactly to the needs of the University Hospital. Because of this, the use of such applications was incomplete and lacking the resources necessary to satisfy the demands of the requesting centre.

To date the Medical Service has attempted to seek solutions by means of databases created by the staff themselves (which give rise to a very simple formulary lacking many of the requirements needed) and even using software created at European level for the storage, analysis and dissemination of valid data concerning the risk of infection in the hospitals of the continent. This application, called HELICS[1] [7], responded to very general aspects that often did not meet the demands of the Hospital Service. It is clear that software oriented to responding to a request based on covering such a large field means that on many occasions it is very difficult to find certain aspects that are necessary at a more particular level.

The main objective of the application demanded is the creation of a system in which it is possible to follow up the clinical history of patients in a simple and clear way.

3 Overview of TENSSION

The application here presented is called "TENSSION" and allows the medical staff to follow up the clinical history of patients in a simple and clear way.

To accomplish this, an individual hierarchical tree has been structured in which each of the patients admitted to the Hospital is stored. Each admission means the patient may pass through three Services (Surgical, Medical and ICU), gathering a daily formulary in which a series of aspects necessary for a complete follow-up of the patient are stored.

Within the structured medical history in the tree are the admissions that the patients has made at the hospital (See Fig 1.), taking the date of admission as the

---

[1] http://helics.univ-lyon1.fr/home.htm

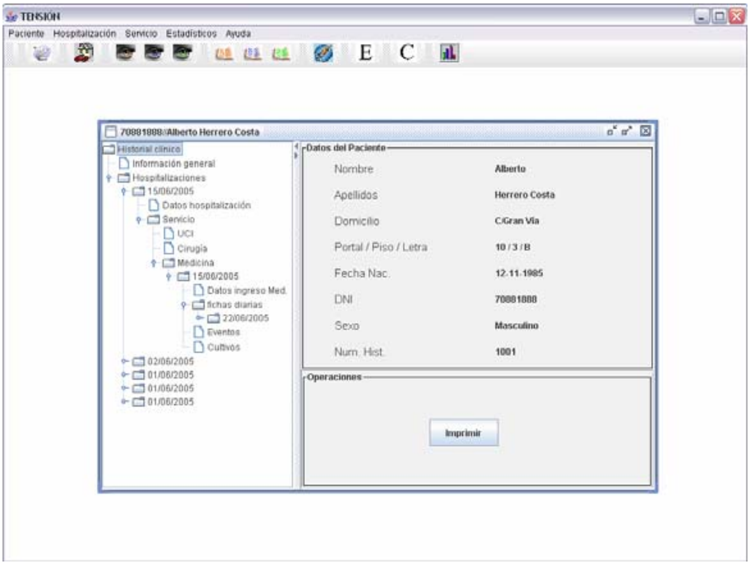


Fig. 1 Admission Patient Information

name of the branch. From each branch mentioned three sub-branches are established in which the three Services through which the patient may pass are differentiated (ICU, Surgical, Medical). In these it is possible to find a new internal tree in which one finds the daily formularies that are filled in on each day the patient is within a given Service. Thus, a hierarchical tree is established whose end is to facilitate the storage of the patient's clinical history and its later visualization during follow-up.

Two types of daily formularies can be distinguished. First, there are those belonging to the Medical and Surgical Services, and then those of the ICU. Both share certain aspects, such as a part in which the instrumentation used is recorded, together with the medication, or a small box in which to record certain aspects of the daily diagnosis (these aspects are necessary to record the daily stay of the patient as completely as possible). In turn, there is a series of risk factors (intrinsic and extrinsic) that may arise, which must be recorded daily for later reference and for performing statistical studies. Accordingly, they are shown in two emerging windows with "check in" buttons where the user can select, or not, those that have emerged.

One important aspect is the incorporation (on the daily formularies of the ICU Service) a system for automatically calculating several test in the medical protocol as the SAPS(Simplified Acute Physiology Score)[8] and the APACHE(Acute Physiology & Chronic Health Evaluation)[9] from a series of data that the user introduces from the keyboard, chosen by the program internally from a series of medical rules, affording the sum of those factors on screen and giving as a result those two significant numbers.

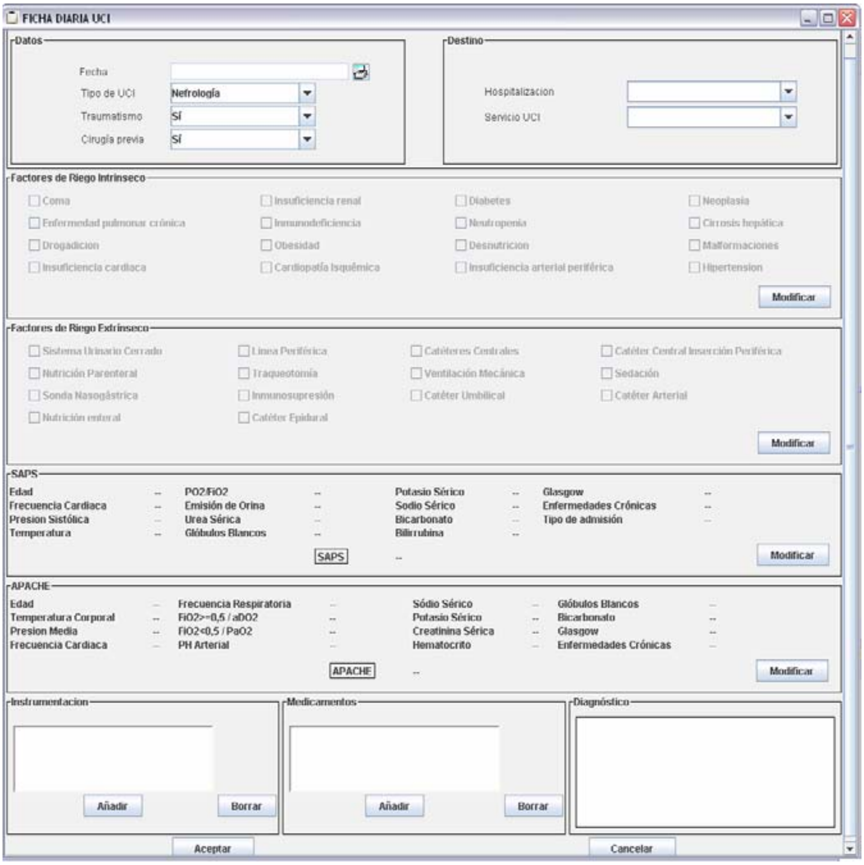


Fig. 2 Daily Formularies

The application provides the admissions in the Surgical Service with the possibility of adding surgical formularies (See Fig. 3) when an intervention is performed. With this, the aim is to automate a work that to date has been performed manually. All the formularies completed will be stored in the interventions branch during each stay of the patient in the Surgical Service.

Another aspect contemplated in the application is the storage of possible unexpected events in a Service that start an alert notification. Each time an event happens, a formulary is made to notify public agencies. Among them are the following:

- EP – TVP
- Falls
- Post-operative hemorrhages
- Bed sores
- Complications from anesthesia

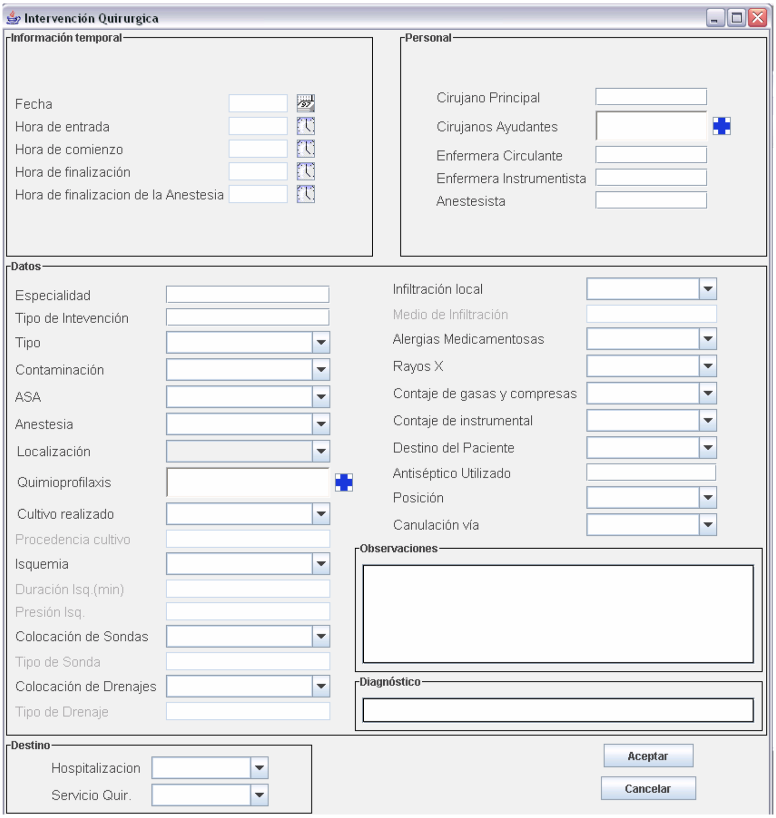


Fig. 3 Surgical Formularies

- Implant complications
- Acute myocardial infarction or heart failure
- Laceration or perforation
- Iatrogenic complications

This factor is important due the evident repercussion that the generalized appearance of some complication may have in the Service; this functionality possibly may affect to a significant number of patients and medical staff.

The formulary includes five different modules that the user has to fill in. We can differ in this modular classification:

1. Fist module: Patient information and precedents.
2. Second module: The injury and its effects.
3. Third module: Period of hospitalization during which the event happened.
4. Fourth module: Principal problems during the assistance process.
5. Fifth module: Causal factors and possibility of prevention.

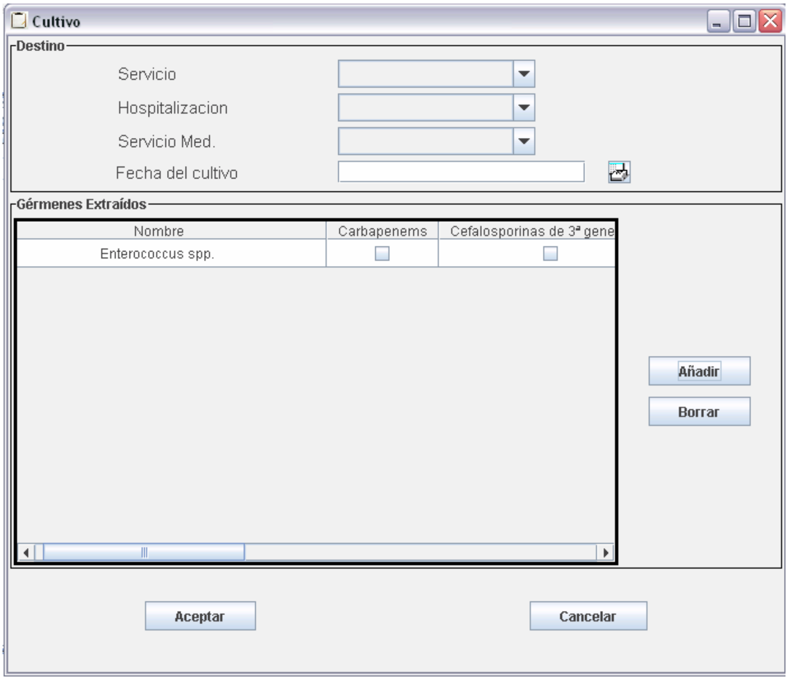


Fig. 4 Window for Recording the Microbiological Cultures

The tool allow to record the microbiological cultures made (see Fig 4), and the bacteria isolated in the rows of a square located in an emerging window with information about the antibacterial agents to which they are resistant.

Architecture

The system has been developed following the view-controller model to make easy the possibility of modifications in the application in relation with requesting agent.

We sought to create a system of computers with an intranet network (See Fig. 5). The physical elements are as follows (Intranet network, Equipment of the Medical Centre, Database and secret files, Administrator and Server). Below we detail each of these elements:

- *Intranet network*. The teams of the Medical Centre will be connected through an intranet network so that they can handle the data from any point on it. In this way, the information will not be concentrated on one computer, which would mean that it would not be possible to access the application at each of the services distributed through the hospital in turn implying the danger that might arise from basing the system on the functioning of a single device for data management.

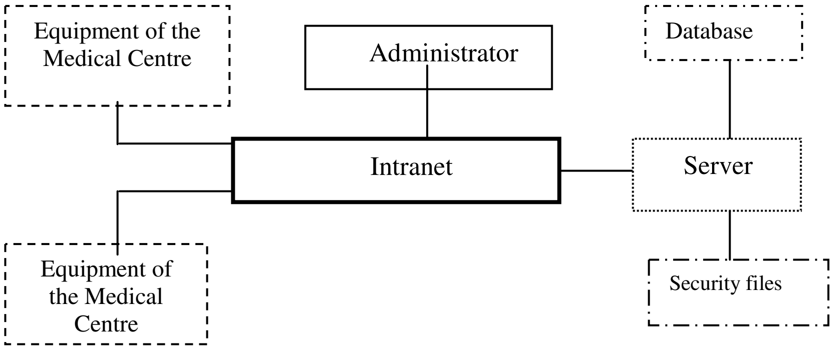


Fig. 5 Window for Recording the Microbiological Cultures

- *Equipment of the medical centre*. This refers to the different machines being run at different places in the hospital. They provide and retrieve information from the server.
- *Databases and secret files*. These contain the information harboured by the application, from which follow-up is accomplished at patient level. This information can be used to perform statistical studies at the various levels of the centre (very important for the assessment and maintenance of services, for individual and collective follow-up, and for the prevention of possible risks that may arise in the centre for different reasons).
- *Administrator*. This is in charge of managing the system. There will be permission to access and modify certain aspects of the system for maintenance purposes.
- *Server*. This attends to the requests made by users with permission to access the system (authenticated users) and the administrator. When necessary, the databases and secret files can be consulted to return the information requested.

Once the physical parts have been detailed, we now focus on the architectural pattern used (the view controller model) that will serve as a mould for the application.

The following parts can be differentiated:

- Views. Responsible for receiving data from the model and showing them to the user.
- Controller. Receives the input events and contains rules for event management.
- Model. Accesses the data storage layer and defines the business rules.

The dependence among layers will always be established in a descending sense and not ascending. In this way layer one view will depend on the two lower ones. In turn layer two (controller) will only depend on layer three, and this latter will be completely independent.

4 Results and Conclusions

The paper presents an application of crucial importance owing to the huge problem of the University Hospital of Salamanca regarding data storage and the follow-up of patients, since to date this activity has been performed manually.

The tool, called "TENSSION" allows an exhaustive follow-up of a patient's stay in the centre, storing each of the admissions, the Services through which he /she passed and a list of daily annotations containing the most important data. It is possible to carry out statistical studies with the data from the centre in a more general way and thus have information about patient evolution and which is critical to generate alerts when possible complications arise. The application allows the medical staff to follow up the clinical history of patients in a simple and clear way. The system can be adapted and upgraded to meet changing needs in order to the success the implementation and acceptance of the computerized patient information system in the Hospital.

References

1. Dudman, J.: Electronic patient records: facts at their fingertips. Health Service Journal 110(5728 suppl.), 9, 10, 113 (2000)
2. Dugas, M., Uberla, K.: MediDOK: an Intranet tool for XML-based data modelling in medicine. Studies in Health Technology and Informatics 22, 983–987 (2000)
3. Espinosa, A.: Implementing an ICU-CIS integrated to an HIS in Latin America. Studies in Health Technology and Informatics 77, 916–920 (2000)
4. Greemes, D.S., Fleisher, G.R., Kohane, I.: Potential impact of a computerized system to report late-arriving laboratory results in the emergency department. Pediatric Emergency Care 16, 313–315 (2000)
5. Knaup, P., Mludek, V., Wiedemann, T., et al.: Integrating specialized application systems into hospital information systems: obstacles and factors for success. Studies in Health Technology and Informatics 77, 890–894 (2000)
6. Modai, I., Rabinowitz, J.: Why and how to establish a computerized system for psychiatric case records. Hospital and Community Psychiatry 44, 1091–1095 (1993)
7. Suetens, C., Morales, I., Savey, A., Palomar, M., Hiesmayr, M., Lepape, A., Gastmeier, P., Schmit, J., Valinteliene, R., Fabry, J.: European surveillance of ICU-acquired infections (HELICS-ICU): methods and main results. Journal of Hospital Infection 65, 171–173
8. Gall, J.-R., et al.: A Simplified Acute Physiology Score for ICU Patients. Crit. Care Med. 12(11) (November 1984)
9. Zimmerman, J.E., Kramer, A.A., McNair, D.S., Malila, F.M.: Acute Physiology and Chronic Health Evaluation (APACHE) IV. Critical Care Medicine 34(5), 1297–1310 (2006)

Dynamic Planning with Bayesian Network Applied in MAS

Juan F. De Paz, Manuel Pablo Rubio, and Angélica González

Abstract. The application of information technology in the field of biomedicine has become increasingly important over the last several years. This paper presents a multi-agent system that incorporates agents with automatic planning capabilities based on the CBP-BDI (Case Based Planning) (Belief, Desire, Intention) architecture. This agent proposes a new reasoning agent model mechanism that can model complex processes as external services. The agents act as Web services coordinators that implement the four stages of the case-based planning cycle. The multi-agent system has been implemented in a real scenario to classify leukemia patients. The results obtained are presented within this paper and demonstrate the effectiveness of the proposed organizational model.

Keywords: Multiagent Systems, Case-Based Reasoning, BDI, Bayesian Network, Case-based planning.

1 Introduction

Biomedical systems have become very important in recent years, automating tedious tasks and providing novel decision support systems in fields such as cancer diagnosis. Many of these systems are based on the use of artificial intelligence and attempt to imitate human behavior; but in general, they focus on very specific tasks such as providing decision support in medical environments (Chua et al. 2008) (François 1992) but are not capable of automated task planning. This article proposes a multi-agent system that can assist in carrying out this type of planning.

There currently exist a number of applied computing techniques [1] [2] that can perform microarray data analysis. These techniques can be selected to carry out various tasks that involve analysis, whereby the actual selection is just as important as the order in which they are applied. Nowadays, there are different approaches such as

Juan F. De Paz · Manuel Pablo Rubio · Angélica González
Departamento Informática y Automática
Universidad de Salamanca
Plaza de la Merced s/n, 37008, Salamanca, Spain
e-mail: {fcofds,mprc,agelica}@usal.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 113–121.
springerlink.com

myGrid [3] [4] that aim to facilitate the process of selecting techniques and ensuring a smooth execution. They base their functionality on the creation of web services that are implemented with the OGSA (Open Grid Services Architecture) [5]. Their main disadvantage, however, is that the user must be responsible for creating the sequence of actions that resolve a specific problems.

This research presents an entirely new perspective for dealing with an architecture capable of modeling biomedical organizations through multi-agent systems. The multiagent system consists of a Virtual Organization in which each of the roles that are carried out in biomedical organizations is modeled by agents, and the collaboration between the roles is represented as interactions between intelligent agents. Furthermore, the system is capable of automatic reorganization, and as such is capable of evolving and adapting to social changes that take place within the organization. Initially, the system was conceived as a dynamic multi-agent architecture and was oriented to KDD (Knowledge Discovery in Databases) as applied to biomedical databases, thus providing an innovative decision support system. The intelligent behavior of the agents is achieved by applying supervised learning techniques, such as Case based Reasoning (CBR) [7] and related systems like Case Based Planning (CBP) [8] with the BDI (Belief, Desire, Intention) models [8].

Previous research [14] produced automated planning mechanisms, but the system was incapable of generating new plans and was limited in its ability to select the optimal plan based on past experiences. As a case study, the automated planning system has been applied in the expression analysis of microarray data. Specifically, it has been applied in a previous case study involving CLL leukemia patients [9].

Section 2 presents the automated planning process. Section 3 describes a case study and finally, Section 4 presents the results and conclusions obtained.

2 Dynaminc Analysis in Multiagent Systems

This study stems from the multi-agent system previously described in [9]. The initial work contained a multi-agent system that was broken down into several layers. The first layer contained the organization agents, which consisted of the laboratory, analysis and doctor agents. The second layer of analysis used a series of web services to carry out different types of information processing. The agents at this level included the preprocessing, filtering, cluster and knowledge extraction agents. Those agents followed the CBP model while the new agents implemented the CBP-BDI model. This made it possible to generate a new plan by actually reusing previous plans, as opposed to simply reusing the best, or most similar, plan from the memory of plans. This process allows new plans to be created automatically. The agents followed the theoretical concepts from the CBP-BDI model as described in [8] [10]. The new system is based on generating plans p_n that are composed of a sequence of actions a_i that can change between the different states e_j. Each state is defined by a set of variables. This function is shown in equation (1).

$$p_n : \underset{e_0}{E} \underset{\rightarrow}{\rightarrow} \underset{p_n(e_0)=e_n}{E} \tag{1}$$

$$p_n(e_0) = e_n = a_n(e_{n-1}) = \cdots = (a_n \circ \cdots \circ a_1)(e_0) \quad p_n \equiv a_n \circ \cdots \circ a_1$$

It is necessary for a reasoning process to be incorporated into the CBP-BDI model so that plans can be generated as a concatenation of actions. In the classic BDI model, this is done by reasoning engines that are based on a set of rules [11] [12]. The reasoning engines need a computational load and are more geared towards generating a new solution, rather than searching for optimal solutions.

In a multi-agent system, a planning process is considered a sequence of actions carried out in a particular order. The actions in this case are the services that are applied during the process. Because of this, actions can be substituted by services. Figure 1 provides a graphical representation of a service plan. As shown, the graph has only one path and contains nodes that are not connected. The path defines the sequence of services from the start node until the end node. The plan described by the graph is defined by the following sequence $(s_7 \circ s_5 \circ s_3 \circ s_1)(e_0)$. e_0 represents the original state that corresponds to Init, which represents the initial problem description e_0. Final represents the final state of the problem e^*.

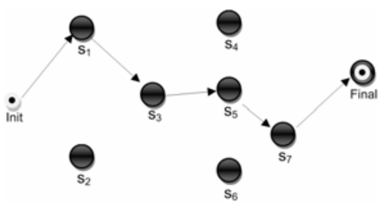


Fig. 1 Planning route for disconnected graph

This way, a CBP-BDI agent works with plans that contain the information associated with the actions that it should carry out, i.e., each agent at the analysis layer defines its own memory of plans with the information it needs. The information required for each of the agents at the analysis layer depends on its functionality. Some agents require executable actions such as service compositions, while others need only select the service that best suits their needs without having to carry out any composition.

CBP-BDI agents use the information contained in the cases in order to perform different types of analyses. As previously explained, an analysis assumes the construction of the graph that will determine the sequence of services to be performed. The construction process for the graph can be broken down into a series of steps that are explained in detail in the following sub-sections (we will focus on one agent in particular within the analysis layer, specifically the filtering agent). Figure 2 illustrates the planning process. In the first stage, the plans that were previously carried out are extracted and categorized as either efficient or inefficient. A new graph is generated for each category, efficient and inefficient, and from each graph the TAN (Tree Augmented Naïve Bayes) model is calculated in order to estimate the probability of execution for each one of the services. According to the probability, a final graph is generated using weights that calculate the optimal route based on either minimizing or maximizing the distance travelled. Finally, the plan is reviewed and if the reviewer determines that the efficiency level is satisfactory, the plan is stored in the memory of cases.

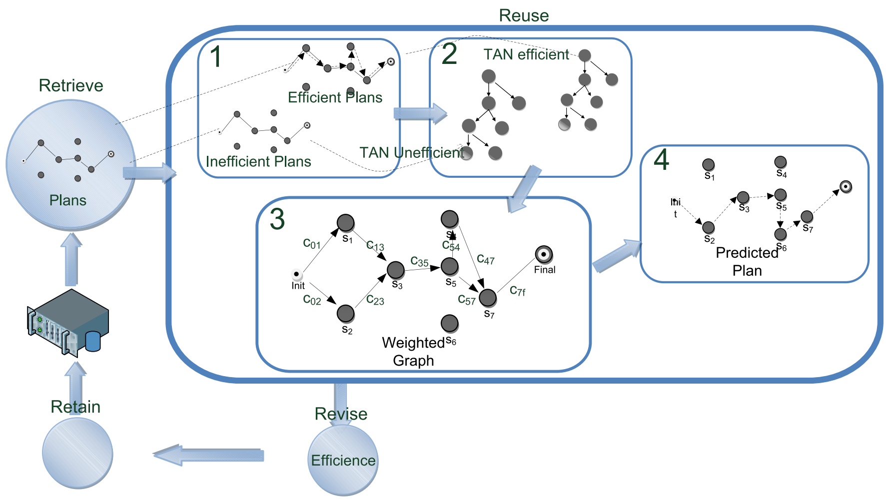


Fig. 2 CBP-BDI planner

The construction of the TAN classifier is based on the plans recovered that are most similar to the current plan, distinguishing between efficient and inefficient plans to generate the model (the tree). Thus, by applying the Friedman-Goldsmidtz [13] algorithm, the two classes that are considered are efficient and inefficient. The Friedman-Goldsmidtz algorithm makes it possible to calculate a Bayesian network based on the dependent relationships established through a metric. The metric considers the dependent relationships between the variables according to the classifying variable. In this case, the classified variable is efficient and the remaining variables indicate whether a service is or is not available. The metric proposed by Friedman be defined as:

$$I(X,Y|Z)=\sum_{x\in X}\sum_{y\in Y}\sum_{z\in Z}P(x,y,z)\cdot\log\left[\frac{P(x,y|z)}{P(x|z)\cdot P(y|z)}\right] \tag{2}$$

Based on the previous metric, the probabilities are estimated according to the frequencies of the data. The Friedman-Goldsmidtz [13].

Using the TAN model, we can define the probability that a particular number of services may have been executed for classes 1 and 0 for the efficient and inefficient plans. This probability is used, together with the probability of execution, to determine the final value for the weight with regards to the quality of the plans recovered. Assuming that the probability of having executed service i for class c is defined as follows $P(i,c)$, the weight of the arcs is defined according to the following equation (3). The function has been defined in such a way that the plans of high quality are those with values closest to zero.

$$c_{ij} = P(j,1)\cdot I(i,j,1)\cdot t_{ij}^{1} - P(j,0)\cdot I(i,j,0)\cdot t_{ij}^{0} \tag{3}$$

$$t_{ij}^{1} = \frac{\sum_{p\in G_{ij}^{1}, s\in G^{1}} (1-(q(p)-\min(q(s)))) + 0.1}{\#G_{ij}^{1}} \tag{4}$$

$$t_{ij}^{0} = \frac{\sum_{p\in G_{ij}^{0}, s\in G^{0}} q(p)-\min(q(s)) + 0.1}{\#G_{ij}^{0}} \tag{5}$$

where:

- $I(i,j,1)$ is the probability that service i for class 1 is executed before of service j.
- $P(j,1)$ the probability that service j for class 1 is executed. The value is obtained based on the Bayesian network. The value is obtained from the Bayesian network, which is obtained from the TAN model.
- $I(i,j,0)$ the probability that service i for class 0 is executed before of service j
- $P(j,0)$ the probability that service j for class 0 is executed. The value is obtained based on the Bayesian network defined in the previous step.
- G_{ij}^{s} is the set of plans that contain an arc originating in j and ending in i for class s.
- G^{s} is the set of plans for class s.
- $q(p)$ is the quality of plan p that also defined the execution time for the plan. The significance depends on the measure of optimization in the initial plan.
- $\#G_{ij}^{s}$ the number of elements in the set.
- c_{ij} is the weight for the connection between the start node j and the end node i.

Once the weights of the arcs have been calculated, the Dijkstra algorithm is used to calculate the optimal route, having previously eliminated the negative arcs by using the sum of the absolute value of the greatest negative value.

3 Case Study: A Decision Support System for Patients Diagnosis

The multiagent architecture presented in this paper has been used to develop a decision support system for the classification of leukemia patients. The case study analyzes the data from CLL leukemia patients and attempts to classify the patients into the three existing subtypes. The data for leukemia patients were obtained with a HG U133 plus 2.0 chip [9].

3.1 Services Layer

The services implement the algorithms that allow the expression analysis of the microarrays. The previous work in [9] provides a complete description of the techniques applied in the system. There are four types of services, however we shall concentrate only on the filtering services, since they are best equipped to carry out the calculations. Filtering Services: Eliminate the variables that do not allow classification of patients by reducing the dimensionality of the data. Three services are used for filtering: (i) Variables with low variability have similar values for each of the individuals, so they are not significant for the classification process. (ii) All remaining variables that follow a uniform distribution are eliminated. The contrast of assumptions followed uses the Kolmogorov-Smirnov test. (iii) The linear correlation index of Pearson is calculated and correlated variables are removed. (iv) Delete the probes that don't have significant changes in the density of individuals.

3.2 Agents Layer

The agents in the analysis layer implement the CBP-BDI reasoning model. To do so, they select the flow for services delivery and decide the value of different parameters based on previous plans made. A measure of efficiency is defined for each of the agents to determine the best course for each phase of the analysis process. At the stage of filtering, the efficiency of the plan p is calculated by the relationship between the proportion of probes and the resulting proportion of individuals falling ill.

$$e(p) = \frac{s}{N} + \frac{i'}{I} \tag{6}$$

Where *s* is the final number of variables, *N* is the initial number of probes, *i'* the number of misclassified individuals and *I* the total number of individuals.

4 Results and Conclusions

This paper has presented a self-adaptive multiagent architecture. The characteristics of this novel architecture facilitate an organizational-oriented approach where the dynamics of a real scenario can be captured and modelled into CBP-BDI agents. The purpose of the tests was to evaluate the efficiency and the adaptability of the approach with respect to the previous version of the system. We started with the set of plans shown in table 1 and applied the set to both systems.

Table 1 Efficiency of the plans. The values in the table represent the services execution order.

Variability (z)	Uniform (α)	Correlation (α)	Cutoff	Efficiency	Class
1	2			0,1931	1
1	2		3	0,1812	1
1				0,2474	0
	1	2	3	0,1882	1
		1		0,2536	0
	1	3	2	0,1921	1
			1	0,2756	0
		2	1	0,1921	1

The original system [14] selected the most efficient plan based on existing plans, but was unable to generate new plans. Table 2 shows the calculated weights according to formula (3). Table 3 shows the flow that was considered most optimal by both systems. The efficiency of the plan proposed by the system is 0.1753, as stated in (6) the efficiency grows as the value decreases.

Table 2 Weighted values of the arcs

c_{01}	c_{02}	c_{03}	c_{04}	c_{12}	c_{2f}	c_{24}	c_{23}	c_{34}	c_{3f}	c_{4f}	c_{43}	c_{1f}
0,166	0,234	0,000	0,006	0,227	0,176	0,185	0,139	0,139	0,424	0,131	0,682	0,071

Table 3 Efficiency of the calculated plans

Case study	Variability (z)	Uniform (α)	Correlation (α)	Cutoff
New System	1	2	4	3
Old System		1	2	3

Figure 3 presents the different CLL subgroups prior de applying Multidimesional Scaling (MDS) to reduce the dimensionality. MDS was applied to the data obtained after applying the automatically calculated filtering process. Figure 3 illustrates that the result obtained can classify the different groups of patients.

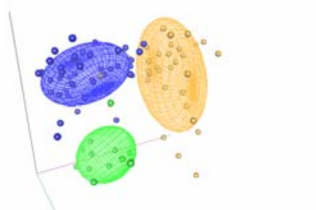


Fig. 3 Low dimensional representation of the automatically filtered CLL Leukemia data

Acknowledgments. This work has been supported by the Ministerio de Ciencia e Innovación project PET2008_0036.

Referentes

1. Riva, A., Carpentier, A.-S., Torrésani, B., Hénaut, A.: Comments on selected fundamental aspects of microarray analysis Comments on selected fundamental aspects of microarray analysis. Computational Biology and Chemistry 29(5), 319–336 (2005)
2. Lee, J.W., Lee, J.B., Park, M., Seuck Heun, S.: An extensive comparison of recent classification tools applied to microarray data. Computational Statistics & Data Analysis 48(4, 1), 869–885 (2005)
3. Stevens, R., McEntireb, G.C., Greenwooda, M., Zhaoa, J., Wipatc, A., Lic, P.: myGrid and the drug discovery process. Drug Discovery Today: BIOSILICO 2(4), 140–148 (2004)
4. Vittorini, P., Michettia, M., di Orio, F.: A SOA statistical engine for biomedical data. Computer Methods and Programs in Biomedicine 92(1), 144–153 (2008)
5. Foster, I., Kesselman, C., Nick, J., Tuecke, S.: The Physiology of The Grid: An Open Grid Services Architecture For Distributed Systems Integration. Technical Report of the Global Grid Forum (2002)
6. Sirin, E., Parsia, B., Wu, D., Hendler, J., Nau, D.: HTN planning for Web Service composition using SHOP2. Web Semantics: Science, Services and Agents on the World Wide Web 1(4), 377–396 (2004)
7. Riverola, F., Díaz, F., Corchado, J.M.: Gene-CBR: a case-based reasoning tool for cancer diagnosis using microarray datasets. Computational Intelligence 22(3-4), 254–268 (2006)
8. Glez-Bedia, M., Corchado, J.: A planning strategy based on variational calculus for deliberative agents. Computing and Information Systems Journal 10(1), 2–14 (2002)
9. Corchado, J.M., De Paz, J.F., Rogríguez, S., Bajo, J.: Model of experts for decision support in the diagnosis of leukemia patients. Artificial Intelligence in Medicine 46(3), 179–200 (2009)
10. De Paz, J.F., Rodríguez, S., Bajo, J., Corchado, J.M.: Mathematical model for dynamic case based planning. International Journal of Computer Mathematics 86, 1719–1730 (2009)
11. Fasli, M.: Interrelations between the BDI primitives: Towards heterogeneous agents. Cognitive Systems Research 4(1), 1–22 (2003)

12. Jonkera, C.M., Snoep, J.L., Treura, J., Westerhoff, H.V., Wijngaards, W.C.A.: BDI-modelling of complex intracellular dynamics. Journal of Theoretical Biology 251(1), 1–23 (2008)
13. Friedman, N., Geiger, D., Goldszmidt, M.: Bayesian Network Classifiers. Machine Learning 29, 131–163 (1997)
14. De Paz, J.F., Rodríguez, S., Bajo, J.: Multiagent Systems in Expression Analysis. In: 7th International Conference on PAAMS 2009. Advances in Soft Computing Series, vol. 55, pp. 217–226 (2009)

Enhancing the Role of Multi-agent Systems in the Development of Intelligent Environments

Davide Carneiro, Paulo Novais, Ricardo Costa, and José Neves

Abstract. The development of Intelligent Environments is a complex challenge. This complexity arises, in part, from the amount of different devices that need to be seamlessly integrated in a common and homogeneous environment, despite the fact of each device having its own characteristics. This heterogeneity of devices is particularly risky when one passes from the specification to the implementation phase, where all unexpected things start to happen. Multi-agent systems are the paradigm par excellence for implementing Intelligent Environments. However, traditionally, agents are only used at the implementation phase. In this paper we propose a new 3 step approach in which agents are used during all the development process, playing undoubtedly a much more preponderant role and making the path from the specification to the implementation a much easier and controllable one, always having in mind the challenges of the development of Intelligent Environments.

Keywords: Multi-agent systems, Distributed Computing, Ambient Intelligence.

1 Introduction

Ambient Intelligence (AmI) [7] has been revealed in the last years as one of the most promising and exiting research fields. In fact, the possibilities that arise from this merging of Ubiquitous Communication, Ubiquitous Computing and Intelligent User Interfaces are what a few years ago only science fiction visionaries could dream on. Ambient Intelligence may even be considered to be a revolution as it completely changes the way we see and we interact with computers, definitively ending with the long-established desktop paradigm.

Davide Carneiro · Paulo Novais · José Neves
CCTC, Departamento de Informática, Universidade do Minho, Braga, Portugal
e-mail: {dcarneiro,pjon,jneves}@di.uminho.pt

Ricardo Costa
CIICESI, College of Management and Technology of Polytechnic of Porto, Felgueiras, Portugal
e-mail: rcosta@estgf.ipp.pt

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 123–130.
springerlink.com

In AmI the most important change is that the user is placed at the center of the system. We have no more to move close to a computer and use it according to its ways. Now, computers come to surround us, disguised behind our common appliances, and constantly interact with us, sometimes without us noticing it, and constantly providing services and taking actions that empower our experience.

However, as in every new technology, there are challenges that need to be addressed [2001]. Ambient Intelligence is not an exception. Most of the challenges arise from the fact that it is built out of many different components. In fact, it is common to find objects as different as sensors, personal devices, regular computers or actuators. Other than physical components, we must also consider the software ones: databases, decision support systems, inference engines or remote services, only to name a few. We cannot fail to remember that each of these components has its own characteristics: a personal device has limited power capability and uses a wireless communication protocol, a sensor uses a dedicated cable and its manufacturer's communication protocol and an actuator may use either the power line or a wireless protocol to communicate.

It is a fact that Multi-agent systems can and have been used to overcome these difficulties. The classical approach is to design a specification of the problem, and then either simulate it or advance right to the implementation phase. It is commonly accepted that simulation is a great tool for the development of such complex systems: it allows for the test and assessment of the system before its actual implementation. Advancing to the implementation phase without at least simulate the desired system may therefore be irresponsible and carry a larger and more costly implementation phase. However, simulation by itself, is not a guarantee of much better results. In fact, when one looks at some current AmI simulation projects [8, 11], it is easy to conclude that the simulation architecture as well as the technologies used, are sometimes so different from the technologies used for the implementation that the moving on to that phase may be almost as costly as doing it without having used simulation.

What we propose is a 3 step process that fully exploits the possibilities of agents during all its phases: specification, simulation and implementation. This will allow, as many other approaches, to develop agent-based Intelligent Environments. However, having in mind the final agent-based architecture since the first stage of development, will allow the development of a better suited specification as well a simulation platform that truly resembles the implemented system, definitively making the implementation a smoother and more controlled process.

2 The Classical Approach

The use of agents for the development of Intelligent Environments constitutes no novelty. It is a paradigm so suited to this task that it is easy to find several projects that implement the services of AmI on top of agents. See for example [9] and [3]. Classically, what is done is to specify a set of agents, each one with well defined roles (e.g. temperature control, presence control, agenda management, power management). The agent paradigm is in fact very suited to do this task.

Agents have the autonomy to take their own decisions towards the achievement of their goals, can communicate with other agents and can even negotiate in order to have some influence on the beliefs of others [10]. Negotiation [5] is in these environments very important as it is vital to solve the common cases of contradictory objectives (e.g. the comfort agent wants to raise the temperature in the room but the energy control agent wants to save energy).

More than that, agents can hide the singularities of the devices they represent. A temperature agent, for example, is the specialist in reading the temperature from a given sensor. No other agent needs to have knowledge about how to read the temperature, all they have to do is to request the temperature value to the temperature agent, over a standard communication protocol.

3 The Three Step Approach

In section 2 we have seen the way that agents have been exploited until the present for the development of AmI. However, it is easy to conclude that agents play only a preponderant role in the implementation phase of the environment. Could they not be used since the beginning of the process? Wouldn't it be possible to develop a simulation platform whose architecture would match the final architecture, thus agent-based? It is our conviction that it is possible and that it significantly improves the process of the development of Intelligent Environments.

In fact, what we propose in this paper is a three step process in which the agent paradigm is present right from the beginning. In the first step the agents and their hierarchical structure are defined. In the second step a simulation platform is built that respects the architecture defined in the first step, with every agent simulating the services it will actually provide when implemented. We will call these agents Virtual Agents (VA). This naming convention is rather misleading as these agents are evidently real agents, living in a real platform, nevertheless acting in a virtual environment shaped after the image of the target environment. It will however be useful for distinguishing between these and the Real Agents (RA), the agents operating in the real environment. The path to the third step is taken when the VA are gradually replaced by the RA: similar agents that do not simulate their services but in fact provide them in the real environment. At some intermediary point there will be, in the same platform, Virtual and Real agents interacting. These two types of agents are actually the same and run in the same platform, thus allowing communication between them. The only difference is that RA work with real components (e.g. sensors, actuators, databases) while the VA works with virtual or simulated ones (e.g. virtual sensors, simulated user).

Step 1: Specification

The first step is to completely specify the problem. There are several main issues to have in mind when designing the specification of an intelligent environment. It is mandatory to know the type of environment that will be implemented (e.g. domestic, hospital, commercial, geriatric). Based on this, the main services to be provided will be depicted and therefore the devices needed to implement these services will also be known. The target type of users is as important as the type of environment, so that the services are personalized. In fact, most of the services need to be adapted according to the context of the user (e.g. role in the environment, age, physical impairments, gender). Having defined these two key issues, work may start on the definition of the agents.

A methodology suited for building complex agent-based systems is called evolutionary development of systems. It consists in starting the development with a reduced number of agents, each one representing a group of services that can be grouped according to some criteria. Each of these agents is then sub-divided into more simple agents, and so on, resulting in a hierarchy that, as we go to lower levels, find more specific and simpler agents [4]. This has several advantages. In one hand, it is simpler to analyze the agents and detect eventual errors or unexpected behaviors as each agent has simple and well defined tasks. In the other hand, it makes it possible to reuse agents and services, resulting in more lightweight and scalable systems.This process of iterative subdivision is also known as compositionality. This concept denotes entities that can be described by the sub-entities that make part of it and the rules used to combine them. In the specific case of an agent-based platform, it can be described by the agents that make part of it and their relations (e.g. social, hierarchical), the sub-agents that make up each main agent and their relations, and so on.

When defining this hierarchical structure of agents, we will have in mind one central idea: the architecture that the hierarchy of the agents defines will be the architecture of the simulation platform and the architecture of the final environment. This means that the simulation platform and the real environment will be compatible and will even be possible for them to coexist.

Step 2: Simulation

The main objective of this second step is to look at the specification defined before and implement all the agents respecting their hierarchical structure. Moreover, each agent will simulate the services needed to implement the roles that were assigned to it. What we achieve is an architecture that matches the one of the final implementation of the system, with the same agents providing the same services, although in a simulated fashion. However, in order animate this simulation, two more agents are necessary. The first one, which we will call the Simulator agent, will be responsible for controlling the simulation-related parameters, namely the velocity of the simulation, its length or the time of the day. The second one is the Environment agent. This agent will simulate the environment in terms of its main characteristics, namely the environmental parameters, the physical properties, the users, the appliances, among others. This will make it possible for the agents of the platform to acquire the data from the environment that they need to take their decisions.

Once the simulation is running a wide range of tests and assessment can be performed on the architecture and their building blocks in a much more controlled environment. There are two main objectives here: to test the communication protocols defined previously and to test the services of the agents and their effects on the simulated environment.

Should any modification be necessary and an agent or the whole architecture can go back to step one, for a better specification, whether it is by better detailing an agent, modifying its role on the architecture, rearranging the hierarchy, among others. We can now make the exercise of imagining doing these changes with the real implementation and the complexity of changing a system that has already been deployed. In fact, using this approach, one gets a simulation platform that is as close as possible to the architecture of the final implementation, which allows detecting situations that would, otherwise, only be visible after the implementation. Another advantage is that this is an iterative process that can be repeated until one reaches the desired architecture, with the desired behavior, and only then, with more confidence, advance to the implementation phase.

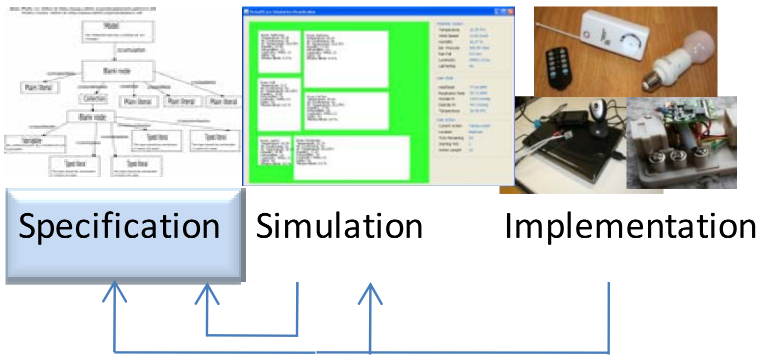


Fig. 1 The three steps of the development process

Step 3: Implementation

The final step consists in implementing the architecture defined and simulated. Traditionally, one would have to implement all the agents and their services to, only then, start making the final tests to the architecture. However, using this approach, it is possible to gradually replace a VA for its equivalent RA, resulting in a much smoother and controlled transition from the simulation to the implementation phase. The only requirement here is that the RA has the same name and the same services signature of the VA it is replacing. This way, compatibility is kept with the remaining agents that eventually use its services. The main advantage here is the possibility of replacing one agent at a time, being possible to perform all the tests to each new RA while he is already integrated in a running platform, eventually a partially simulated one. If an error is found on a new RA, it can be removed from the architecture and again replaced by its equivalent VA, ensuring that the architecture will keep running so that other tests can be performed to other agents. This can be done individually to each new RA, maximizing the confidence on the final architecture and making it easier to isolate errors or wrong behaviors.

Hence, there is not an explicit point at which step two ends and step three starts. Instead, they overlap (Figure 1). Step three starts when the first VA is replaced for its equivalent RA and step three ends when the last VA is replaced. Evidently, during this process, if necessary, the process can go back to step one, for some adjustments. This constitutes no major problem to a modular agent-based architecture.

4 Our Experience: The VirtualECare Project

The ideas depicted here are the result of the experience of this team in AmI projects. The most recent one, the VirtualECare project, is the soundest example in which this three step process has been applied with success. The VirtualECare project [6] is built on top of a Jade-based Multi-agent System in support of computational systems that seek to monitor and provide personalized health care services to its users or patients.

This way, our work in VirtualECare began by defining the main high level agents. The first configuration was composed of six agents: *Supported User*, *Environment*, *Group Decision*, *Call Service Center*, *Call Care Center* and *Relatives*. For the sake of space, we will not detail all the agents and their complete hierarchy here. We will, instead, focus on the Environment agent, the one that is undoubtedly more interesting given the ambit of this paper (Figure 2).

When we thought of this agent, we had in mind that we needed an agent that could provide all the important information about the environment, namely values from the sensors, the intrinsic characteristics of the environment or the appliances

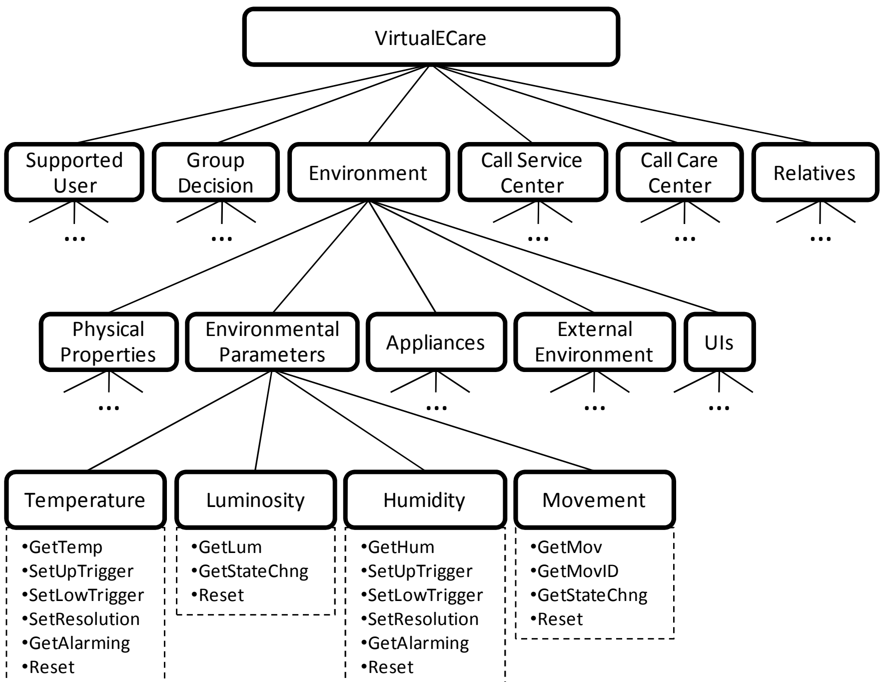


Fig. 2 A detail of the agent's hierarchy that make up VirtualECare environment. The leaf agents are the simple agents with their roles well defined in terms of the services they provide.

present. Evidently, this would result in a very complex agent. Therefore, it has been sub-divided into five agents. The *Physical Properties* agent deals with the physical properties of the environment such as the geographic distribution of the rooms, its insulation and its geometry, among others. The *Environmental Parameters* agent deals with all the sensorial information. The *Appliances* agent is responsible for communicating and issuing orders to every compatible appliance in the environment. The *External Environment* agent was made responsible of sensing the outside environment that surrounds the setting, namely in terms of the temperature, rainfall or luminosity. The *UIs* agent deals with all the interfaces with the user. The same process of sub-division was iteratively applied to the resulting agents as needed in order to minimize their complexity, as visible in Figure 2.

Having finished the Specification phase, the team advanced into the Simulation phase. In this phase a fully configurable simulation platform [2] has been developed and its architecture interactively tested and improved. This architecture, as stated before, matched the specification defined and the final architecture. Once we achieved a stable hierarchy of intercommunicating agents, we advanced into the implementation phase. This last phase consisted in developing the Real Agents. Taking as example the *Temperature* agent, we modified de code of the VA so that instead of simulating a temperature value, it actually read the temperature value of a given sensor. All the structure of the agent remained the same, including the signatures of the services in order to maintain the compatibility. The only thing left was to replace the RA for the VA.

This same process has been applied to other agents and has revealed to be a rather smooth one. At this moment, as we are not yet in possession of all the hardware required for implementing such architecture, we have a hybrid architecture that mixes VA with RA. It is, however, a stable and polyvalent one that allows us to develop and test AmI services, which would otherwise be impossible.

5 Conclusions

In this paper we have presented a new approach for the development of Intelligent Environments based on the agent paradigm. It is a three step process that has as main objective to fully exploit the capabilities of this paradigm from the very beginning of the development. It also constitutes a way of developing a simulation that actually matches the configuration of the final environment, which allows testing not only the agents but the architecture itself. This configures definitively a step ahead as the simulation platforms developed nowadays generally focus on the services themselves and completely forget the architecture, resulting in a simulation that is much different from the implementation. As we concluded during the development of the VirtualECare environment, this approach results in a development process that is faster, simpler, smoother and more homogeneous.

References

1. Aarts, E., Roovers, R.: IC design challenges for ambient intelligence. In: Proceedings of the conference on Design, Automation and Test in Europe, vol. 1 (2003)
2. Carneiro, D., Costa, R., Novais, P., Neves, J., Machado, J., Neves, J.: Simulating and Monitoring Ambient Assisted Living. In: Proceedings of the ESM 2008 - The 22nd annual European Simulation and Modelling Conference, Le Havre, France (October 2008)
3. Fraile, J.A., Bajo, J., Pérez, B., Sanz, E.: HoCa Home Care Multi-Agent Architecture. In: International Symposium on Distributed Computing and Artificial Intelligence (2008)
4. Jennings, N., Faratin, P., Lomuscio, A., Parsons, S., Wooldridge, M., Sierra, C.: Automated Negotiation: Prospects, Methods and Challenges. Group Decision and Negotiation (2001)
5. Kraus, S.: Negotiation and cooperation in multi-agent environments. Artificial Intelligence 94(1-2), 79–97 (1997)
6. Novais, P., Costa, R., Carneiro, D., Machado, J., Lima, L., Neves, J.: Group Support in Collaborative Networks Organizations for Ambient Assisted Living. In: Oya, M., Uda, R., Yasunobu, C. (eds.) Towards Sustainable Society on Ubiquitous Networks. IFIP. Springer, Heidelberg (2008)
7. Riva, G., Vatalaro, F., Davide, F.: Ambient Intelligence. IOS Press, Amsterdam (2005)
8. Shirehjini, A.A.N., Klar, F.: 3DSim: rapid prototyping ambient intelligence. In: Proceedings of the 2005 joint conference on Smart objects and ambient intelligence: innovative context-aware services: usages and technologies, Grenoble, France, pp. 303–307 (2005)
9. Tapia, D.I., Bajo, J., De Paz, F., Corchado, J.M.: Hybrid multiagent system for Alzheimer health care. In: Proceedings of HAIS (2006)
10. Wooldridge, M., Jennings, N.R.: Intelligent Agents: theory and practice. The Knowledge Engineering Review, 115–152 (1995)
11. Youngblood, M., Cook, D.J., Holder, L.B.: Seamlessly engineering a smart environment. In: 2005 IEEE International Conference on Systems, Man and Cybernetics, vol. 1 (2005)

Towards Distributed Wireless Intelligent Sensor Networks

J.A. Fernández-Prieto, J. Canada-Bago, M.A. Gadeo-Martos, and J.R. Velasco

Abstract. Over the past few years, approaches including artificial intelligence models into Wireless Sensor Networks have gained much attention within the research community. However, little attention has been paid to integrate Fuzzy Rule Based-Systems into Wireless Sensor Network. Since traditional approaches of Fuzzy Rule Based-Systems cannot be applied, due to limited resources that the sensors nodes have, the principal aim of this work is to design a distributed knowledge based Wireless Sensor Network, where each node of the network executes an adapted Fuzzy Rule Based-System. This aim has been divided into the followings: 1) to design a Fuzzy Rule Based-System adapted to a sensor node, 2) to design an application protocol which allows transmitting knowledge bases to sensors nodes and 3) to simulate a particular scenario with network simulator ns-2. The performance of a Sun SPOT sensor has been evaluated and a comparison to other devices is reported. The results show that a Sun SPOT sensor can execute properly a Fuzzy Rule Based-System adapted to it, and that it is possible to create a useful distributed knowledge based Wireless Sensor Network.

Keywords: Intelligent sensor, Fuzzy Rule Based-Systems, ns-2, WSN, WISN.

1 Introduction

Wireless Sensor Networks (WSNs) [8] have become an enabling technology for a wide range of applications, such as intelligent buildings, intelligent agriculture, environmental control, health care, disaster relief operations, etc. The traditional WSN architecture consists of a large number of sensors nodes which are densely deployed over an area of interest. These nodes can be conceived as small

J.A. Fernández-Prieto, J. Canada-Bago, M.A. Gadeo-Martos
Telecommunication Engineering Department, University of Jaén Polytechnic School
Alfonso X El Sabio 28, 23700 Linares, Spain
e-mail: {jan,jcbago,gadeo}@ujaen.es

J.R. Velasco
Department of Automatic, University of Alcalá Campus
Universitario 28871 Alcalá de Henares, Spain
e-mail: juanra@aut.uah.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 131–138.
springerlink.com

computers, extremely basic in terms of their interfaces and their components [1], having limited battery, reduced memory and processing capabilities.

Intelligent (or smart) sensors are more sophisticated than traditional sensors as they gather, analyse and transmit data. A definition of intelligent sensor was given by Brignell [4], as one which "modifies its internal behaviour to optimise its ability to collect data from the physical world and communicate them in a responsive manner to a host system". The development of intelligent sensors depends on advances in hardware (i.e. measurement technology) and advances in software (i.e. processing technology).

Some techniques have been developed to implement the capacity modifying its internal behaviour. Hence, in order to implement intelligence into sensors, different kinds of artificial intelligence systems can be used: artificial neural network, fuzzy logic and a hybrid neuro-fuzzy system. Benoit et. al [3] described a modeling of intelligent sensor and how the fuzzy logic approach can be used. In WSNs, different embedded artificial intelligence models have been used [2][3][9][10]. These networks are known as Wireless Intelligent Sensors Networks (WISNs) [13].

However, little attention has been paid to integrate Fuzzy Rule Based-Systems (FRBSs) into WSNs, where each sensor node of the network can execute a FRBS which consists of a knowledge base (KB) composed of a set of IF-THEN fuzzy rules, fuzzyfication and defuzzyfication interfaces, and an inference engine [6].

Since traditional approaches of FRBS can not be applied, due to limited resources that the sensors nodes have, this work proposes a new distributed approach in which sensors nodes can execute a small FRBS adapted to them. In order to create a knowledge based distributed WSN, this aim has been divided into the followings: 1) to design and implement a small FRBS adapted to a sensor node; 2) to design and implement an application communication protocol which allows transmitting KBs to the sensors nodes from a PC; and 3) to simulate a particular scenario with network simulator ns-2 [16]. Hence, the proposed system has been designed in a distributed way in order to separate some functions which traditionally are integrated into the inference engine of a FRBS. The knowledge based distributed WSN could be used in distributed complex problems in which imprecise and uncertain knowledge exists.

The remainder of the paper is organized as follows. Section 2 provides the general structure of the proposed system. Results are reported in Section 3 and finally some conclusions and future work lines are drawn in Section 4.

2 Structure of the System

One of the most important aspects of sensors is that their resources are highly constrained. As consequence, it is not possible to utilize directly the traditional approaches of FRBS and it is necessary a new distributed approach in which sensors could execute a small knowledge based system.

The proposed system has been designed in a distributed way. Figure 1 shows the general structure of the system, which is composed of: **a)** one **Personal Computer,** with a base station gateway, where the users edit the KBs using linguistic labels (variables, fuzzy sets and rules); **b) wireless sensor network**, based on Sun SPOT (Sun Small Programmable Object Technology) sensors [14], which are specially built to run Java programs, with a 180 MHz 32-bit ARM920T processor, 512K RAM and 802.15.4 radio; **c) an application protocol,** which allows transmitting and updating KBs to the sensors nodes from the PC; and **d) a FRBS adapted to the sensors**, which can infer the output by means of an inference engine and its KB.

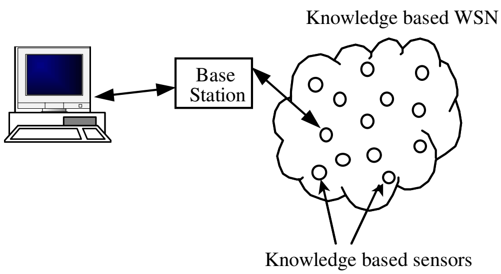


Fig. 1 General structure of the system

2.1 Fuzzy Rule Based System Adapted to the Sensors

Due to the sensor constrains, the traditional approaches of FRBS, which need a large amount of computer resources, cannot be directly applied in sensors. Therefore, it is necessary to design a FRBS adapted to them, reducing these needed resources. The proposed system is based on the basic structure of Mandani FRBS, introducing some modifications in the inference engine. Figure 2 shows the structure of the fuzzy ruled-based sensor which consists of input and output interfaces with scaling functions, fuzzyfication and defuzzyfication interfaces, knowledge base, communication module, and an adapted inference engine.

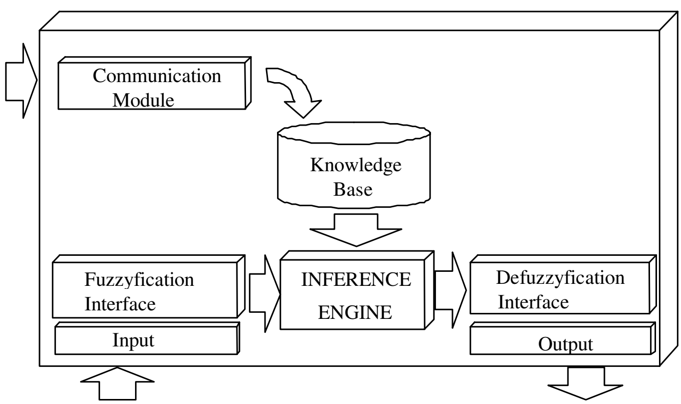


Fig. 2 Structure of the knowledge based sensor

The KB, which is composed of data and rule bases, includes variable definitions, fuzzy sets defined for each variable and rules. After the variables are defined, fuzzy sets are associated to each variable. Each fuzzy set is defined by means of four points which are composed of a pair of real numbers in [0,1], allowing users to utilize triangular or trapezoidal shapes. The KB includes a set of IF- THEN rules, in which every rule has one antecedent with various prepositions and one consequent. The antecedent and consequent relate a variable and a fuzzy set defined in the variable.

The main constraints of the sensors nodes depend on energy resources and computational capacity. Hence, we have introduced some modifications in the traditional approach of FRBS in order to reduce: a) the computational burden and b) the battery consume.

To reduce computational burden, we have designed:

1. A distributed approach in which sensors nodes execute a small but complete FRBS adapted to them.
2. An inference engine which works with numerical values instead of linguistic labels. Therefore, knowledge base elements, such as fuzzy sets, variables, and rules, do not utilize linguistic labels. However, users continue using labels because the man machine interface (in the PC) let them introduce the knowledge using linguistic terms. After that, knowledge elements are translated to numerical values with which sensors operate.
3. A FITA (First Infer Then Aggregate) inference engine which requires less computational burden.

In order to reduce the battery consumption, prolonging the battery life:

1. The sensors nodes have been configured in a shallow sleep mode.
2. The application protocol has been designed in an incremental way, i.e, firstly the entire KB is sent from the PC to each sensor; later, only a part of it (variables, fuzzy sets, etc.) can be updated.
3. The sensors nodes do not transmit data continuously. They process the data and only remarkable results are sent.

2.2 Application Communication Protocol

The general protocol stack used by the sensor nodes is shown in Figure 3. The protocol stack consists of the physical layer, data link layer, network layer, transport layer and application layer [1]. The IEEE 802.15.4 standard [7] has been designed for low power and low data rate WSNs and focuses on specification of the two lowest layers of protocol stack: the physical layer (PHY) and medium access control (MAC) .

Figure 4 shows the specific Sun SPOT network stack protocol. Sun SPOTs are equipped with the IEEE 802.15.4 compliant radio to perform wireless communication. In order to provide routing, meshing and fragmentation, the Sun SPOT stack relies on the LowPAN protocol [11]. Multi-hop connectivity can be accomplished by a sophisticated routing protocol for ad hoc network: AODV [12].

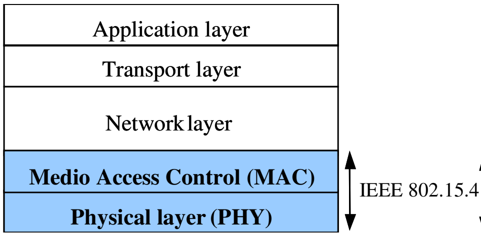


Fig. 3 General sensor network stack

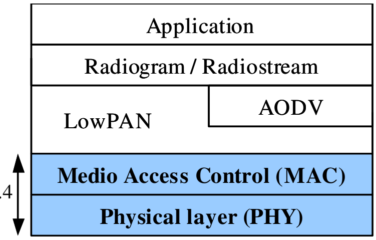


Fig. 4 Sun SPOT network stack

In the transport layer, the protocols which allow Sun SPOT applications to access the network are: the radiogram protocol (similar to UDP) and the radiostream protocol (similar to TCP).

The application protocol has been designed in the highest layer and implemented on the Sun SPOTs and on the PC to transmit entire KBs or a part of them (variables, fuzzy sets, etc.). The KBs are edited and generated in the PC, transmitted to sensors and executed in sensors. Initially, the Sun SPOTs are configured in a shallow sleep mode, with radio and sensor board on (current draw ≈ 46 mA).

The scenarios in which distributed fuzzy reasoning could be applied are those where imprecise and uncertain knowledge exists and where the sensors nodes can work autonomously. For example, in [5] we have shown a control system of plagues in the culture of the olive tree as application of a knowledge based WSN. The development of the olive tree fly is strongly related to the conditions of temperature and humidity of the environment. Every sensor node uses these variables as input and generates an alert status which shows the risk of the plague appearance. If the alert status surpasses a threshold level, the suitability of insecticide treatments applications should be evaluated. This is a perfect scenario where the distributed fuzzy reasoning can be applied.

3 Results

In this section, we show the results obtained. Firstly, we present the performance of a Sun SPOT sensor when it executes the FRBS implemented; and secondly, the simulation results, using ns-2, are shown.

Sun SPOT Performance. The initial KB implemented was composed of 3 IF-THEN fuzzy rules, 2 input fuzzy variables and 1 output fuzzy variable, having 5 membership functions each variable. The Sun SPOT executed 983 inferences per second, i.e. 2950 rules/s, so the Sun SPOT reaction time was of 1.01 ms. Later, the number of fuzzy rules was increased progressively until 15 rules, obtaining a diminution of performance. In this case, with 15 rules, the Sun SPOT executes 223 inferences per second, having a reaction time of 4.4 ms. For instance, in a control system of plagues in the culture of the olive tree [5], it is necessary to do one inference each 15 minutes, so the reaction time is more than sufficient.

In order to compare this results with the obtained in others devices, we executed the FRBS implemented with the initial KB in: **a) another sensor**, which hardware is based on a 80C552 micro controller with 11.059 MHz 8-bits, 256 bytes of internal data memory and 32 KB of external RAM. The sensor executed 30 inferences/sec (90 rules/s), so the sensor reaction time was of 33 ms.; and **b) a Personal Digital Assistant (PDA)** which is based on a 300 MHz 32-bits S3C244XL processor, 64 MB RAM. The PDA executed 6098 inferences/s (18293 rules/s), so the reaction time was of 0.17 ms.

Simulation Results. The application protocol has been implemented in the ns-2.33 release of the ns-2 network simulator, configuring ns-2 nodes with the same characteristics of Sun SPOT sensors. For this purpose, we have used the transceiver characteristics of the commercially available CC2420 IEEE 802.15.4 radio [15]. The CC2420 is IEEE 802.15.4 compliant and operates in the 2.4 GHz to 2.4835 GHz ISM unlicensed bands. The transmit output power can be set from –24dBm to 0dBm, and the receiver sensitivity is –90dBm.

The thresholds which ns-2 uses to determine whether one packet is received correctly by a receiver sensor are the following: the *Receiver Threshold* , the *Carrier Sensing Threshold* and the *Capture Threshold*. The scenario characteristics and the specific transceiver parameters used in the simulation are summarized in Table 1.

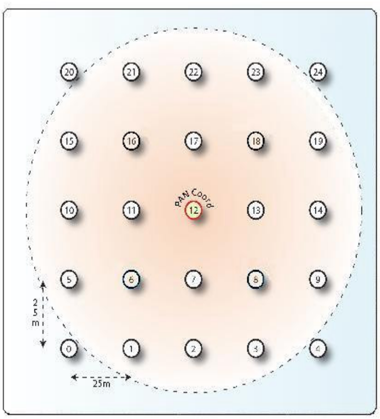


Fig. 5 Network topology

Table 1 Parameters used in the simulation

Simulation area	100 m x 100 m
Number of nodes	25
Pan C. Location	Middle of area
Transmission Range	70 m
Routing protocol	AODV
Link Layer	802.15.4
Simulation Time	40 s
Transmit out power	1mW (0dBm)
Receiver Threshold	-77dbm
Carrier Sensing Th.	-90 dbm
Capture Threshold	10 dB

The model to represent the radio propagation properties of the environment has been the known as free space model. The network topology used in the experiments is shown in Figure 5. The PC with the base station is located in the middle of the area, designated as PAN Coordinator. The KB has been sent from the PAN Coordinator to the sensors sequentially. The PAN Coordinator has one TCP agent for each sensor to transmit the KB, and each sensor was configured as a TCPSink agent. The KB used to carry out the simulation was composed of 15 rules. Figure 6 shows the evolution of the delay when the KBs are transmitted.

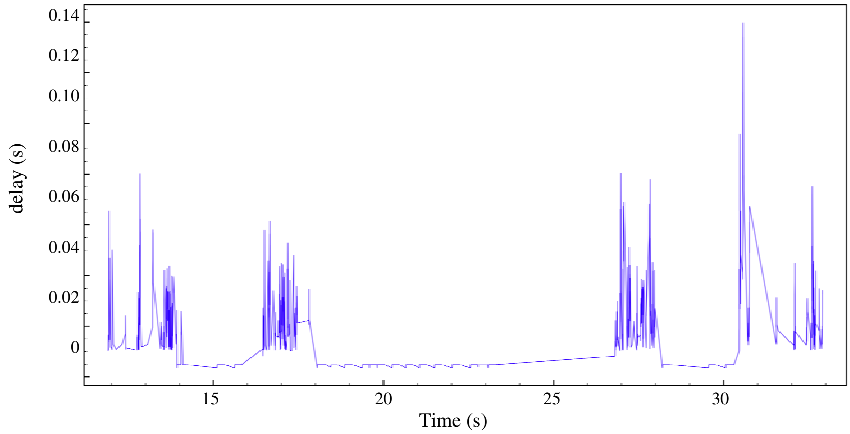


Fig. 6 Evolution of the delay

As can be observed in Figure 6, the graph is symmetric due to the location of the PAN Coordinator, existing four periods of time where there is a delay higher. These periods of time correspond with the sensors allocate around the corners, where a higher number of hops is necessary to access them. During the transmission, the throughput remains around 15 Kbytes/sec (120Kbps) which is a value lower than the theoretical value given in the IEEE 802.15.4 standard (250 Kbps). The results indicate that it is possible to create the distributed knowledge base WSN.

4 Conclusions

In this paper, we have presented a distributed knowledge based WSN, where some functions which traditionally are integrated into the inference engine of a FRBS have been separate, which allows sensors to execute an adapted FRBS. Results have shown the effectiveness of sensors to support a knowledge based system and the distributed knowledge based WSN can be used in distributed complex problems in which exist expert knowledge with uncertainty and vagueness.

Nowadays, the application protocol is being tested in a real WSN testbed which is composed of 10 sensors. In these conditions, the KBs are transmitted correctly. Our future work will be devoted to test the proposed system on a bigger real WSN in order to compare simulations and real results. Besides, it will be applied to a control system of plagues in the culture of the olive tree and an environmental monitoring system.

Acknowledgements. This work has been partially supported by University of Jaén and Caja Rural de Jaén (project UJA 08 16 18).

References

1. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cyirci, E.: Wireless sensor networks: A survey. Computer Networks 38(4), 393–422 (2002)
2. Averkin, A.: Soft Computing in WSNs. In: Proc. of the EUSFLAT, pp. 387–390 (2007)
3. Benoit, E., Dapoigny, R., Foulloy, L.: Fuzzy-Based Intelligent Sensor: Modelling, Design, Application. In: Proceeding of the 8th IEEE International Conference on Emerging Technologies, pp. 401–407 (2001)
4. Brignell, J.E.: The future of intelligent sensors: a problem of technology or ethics? Sensors and Actuators A 56, 11–15 (1996)
5. Canada-Bago, J., Gadeo-Martos, M.A., Fernandez-Prieto, J.A., Velasco, J.R.: Poster Abstract: A Knowledge Based Wireless Sensor Network. In: Roedig, U., Sreenan, C.J. (eds.) EWSN 2009. LNCS, vol. 5432. Springer, Heidelberg (2009)
6. Cordón, O., Herrera, F., Hoffmann, F., Magdalena, L.: Genetic Fuzzy Systems: Evolutionary tuning and learning of fuzzy knowledge bases. Advances in fuzzy systems – Applications and theory, vol. 19. World scientific Publishing, Singapore (2001)
7. IEEE 802.15.4 Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications for Low-Rate Wireless Personal Area Networks (2006)
8. Karl, H., Willig, A.: Protocols and Architectures for Wireless Sensor Networks. John Wiley & Sons, Chichester (2005)
9. Karlsson, B.: Intelligent Sensor Networks - an Agent-Oriented Approach. In: Workshop on Real-World Wireless Sensor Networks (2005)
10. Kulakov, A., Davcev, D.: Tracking of unusual events in wireless sensor networks based on artificial neural-networks algorithms. In: Proc. of the International Conference on Information Technology: Coding and Computing. IEEE, Los Alamitos (2005)
11. Montenegro, G., Kushalnagar, N., Hui, J., Culler, D.: RFC 4944: Transmission of IPv6 packets over IEEE 802.15.4 networks. IETF (2007)
12. Perkins, C., Belding-Royer, E., Das, S.: Ad hoc On-Demand Distance Vector (AODV) Routing, RFC 3561 (2003)
13. Shan, Q., Liu, Y., Prosser, G., Brown, D.: Wireless intelligent sensor networks for refrigerated vehicle. In: Proceeding of the 6th IEEE Circuits and Systems Symposium on Emerging Technologies, vol. 2, pp. 525–528 (2004)
14. Sun Microsystems. Home of project Sun SPOT, `http://www.sunspotworld.com`
15. Texas Instruments. 2.4 GHz IEEE 802.15.4/Zigbee-ready RF transceiver (rev. B)
16. The network simulator - ns-2 (2002), `http://www.isi.edu/nsnam/ns/`

Context-Aware Agents for Vehicular Networks: An Aspect-Oriented Approach

Mercedes Amor and Lidia Fuentes

Abstract. Agent technology plays an important role in the construction of context-aware AmI environments, providing the means to develop intelligent, proactive, autonomous, and interacting systems, able to adapt their behaviour according to some contextual conditions. However, the development of modular context-aware agents has been a challenge to software engineers, since context-awareness use to have influence in several components of the agent architecture, increasing their complexity and making them difficult to evolve and reuse. In order to improve the modularization and development of context-aware agents for AmI applications, this work proposes the application of aspect-orientation to separate all the issues related to context-awareness inside the agent software architecture. Specifically, our approach focuses on the modularization of context-awareness as aspects for Malaca software agents. The application of aspect-orientation leads to separate in aspects all the agent functions related to context-awareness, such as context acquisition and context-dependent behaviour, improving the internal modularization of context-awareness in the agent architecture and making it able to adapt dynamically to different contexts.

Keywords: context-awareness, AOSD, agent architecture, AmI, VANET.

1 Introduction

In essence, Ambient Intelligence (AmI) refers to a digital environment that proactively, but sensibly, supports users in activities of their daily lives in different application domains [1]. AmI puts forward the criteria for the design of intelligent environments, and introduces a set of design challenges that are not present in traditional desktop computing. In particular, it requires context-aware applications that are capable of operating in highly dynamic environments, placing minimal demands on user attention, behaving differently depending on the execution context, anticipating to users' needs, and adapting to changes in the environment.

Mercedes Amor · Lidia Fuentes
Depto. Lenguajes y Ciencias de la Computación
Universidad de Málaga
e-mail: {pinilla,lff}@lcc.uma.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 139–146.
springerlink.com

In AmI environments, agents are a good way to model, simulate, and represent meaningful entities such as rooms, cars, or even users. Context-aware agents are mainly characterized by sensing the environment and adapting its behaviour according to some contextual conditions. Hence, context-aware agents are endowed with appropriate mechanisms and abilities to gather and manage information from the environment, the users, and their activities. Some efforts have successfully introduced context-aware agents for developing AmI applications [3, 5]. However, the incorporation of context-aware concerns adds complexity to the design and implementation of the agent. Context-awareness is usually implemented in agents using traditional OO-based mechanisms, which leads to a complex design where high reusability and maintainability are impracticable [6, 9]. The problem is that context-awareness data and functions are typically scattered across several system modules, producing tangled representations that are difficult to maintain, reuse and evolve.

The main contribution of this work is the use of Aspect-Oriented Software Engineering (AOSD) to facilitate the development of context-awareness concerns in agents. The application of AOSD principles and mechanisms aims at providing improved modularization and composition techniques to appropriately handle context-awareness. Specifically, we will focus on the development of context-related aspects for Malaca [7, 8], an aspect-oriented agent architecture. We present how to modularize context-awareness in aspects dealing with context acquisition and the adaptation of the context-dependent behaviour. These aspects are woven dynamically with context-dependent components of the agent, avoiding including explicit context dependencies (data or functions) in these components. Our main case study is an approach for vehicular ad-hoc networks (VANETs). VANETs applications provide context-aware services to vehicle passengers, where context is mainly determined by the information provided by sensors, vehicles, and users. In our case study, the context-aware agent recommends gas stations based on driver preferences and the current location, fuel level and the speed of the vehicle.

The paper is structured as follow: Next section briefly describes related work, while section 3 introduces the basis of VANETs and brief description of the case study. Section 4 copes with the design and implementation of context-awareness aspects for Malaca agent. Finally, section 5 summarizes the main contributions of our approach and outlines our future work.

2 Related Work

There are many ongoing research efforts that take advantage of aspect-oriented solutions for context-awareness [9, 10] and some of them applies their results to AmI application domains. Actually, [10] shows the benefits of separating such contexts in aspects instead using a simple OO approaches. However, as pointed before, none of these approaches was applied to agent technology.

On the other hand many agents-based approaches provide context-aware services in different application domains, such as traffic and transportation [3] and smart spaces [4]. These approaches use different technologies for dealing with context-awareness, but none of them pay special attention to separate the context awareness property to improve the agent modularization. Also, they are developed for the JADE agent platform, which imposes a coupled internal modularization of the agent application-dependent functions in classes named Behaviours. JADE follows a task-oriented approach, so data and functions related with handling a external event or performing a specific task are included in the same implementation class (of type Behaviour[8]). This means that context-awareness data and functions are spread in different classes of the agent implementation (in those functions which are context-dependent), and tangled with functions concerning to other agent properties. This means that context-awareness concerns are not properly modularized, making difficult their maintenance, evolution, and reuse. As can be seen, the considered approaches focus in providing solutions which are independent of the target agent implementation platform. This makes our approach an enhanced alternative for the design and implementation of context-awareness in MAS and AmI.

3 Case Study: A Refuel Recommender Service for VANETs

A Vehicular Ad hoc Network (VANET) is a special kind of mobile ad-hoc network composed of mobile nodes (mainly vehicles). The specific properties of VANETs allow the development of new services, mainly related to safety and comfort. Safety applications increase the security of passengers by exchanging safety relevant information. Comfort-related applications improve passenger comfort and traffic efficiency. Examples for this category are: traffic information, weather information, gas station or restaurant location, price information, and even Internet access. In a VANET, each vehicle is equipped with a computing device, a short-range wireless interface, a GPS (Global Positioning System) receiver, and different internal sensors, which provide information regarding the status of the vehicle. With these features, agents are a good way to model, simulate, and represent meaningful entities of VANETs. A MAS for VANETs encompasses two different agents: agents inside a vehicle (vehicle agents), and agents representing services and road signs. The former agents run on board to provide safety and comfort-related services to the vehicle passengers. These agents use to be context-aware to deal with the reception of events from different information sources (such as the user and internal sensors), and are able to adapt their behaviour to current context. The service agents represent the traffic signposting and road services (such as gas stations, cafeterias).

Our main case study is a context-aware agent for VANETs which decides when and where the vehicle has to refuel. In a specific scenario, when the agent detects that the level of fuel gas is running low, it decides to refuel. Additional context information, such as vehicle speed, is taken into consideration to make this decision (fuel consume is higher when the speed is low, and it should be acceptable to stop when the tank level is low, and it is not in reserve if the speed is low). Vehicle agents interact with service agents to receive information about gas stations in the proximity. When refuel is needed, autonomously, the agent chooses a gas station meeting user preferences (i.e. a specific gas station chain or gas stations with the lower fuel price).

4 A Context-Aware Agent in Malaca

This section focuses on the design and implementation of the context-aware agent using the Malaca approach.

4.1 Design of a Context-Aware Agent Using Aspect-Orientation

A complete description of the Malaca agent model can be found in [7,8]. The main feature of the internal architecture of a Malaca agent is that combines Component-Based Software Engineering (CBSE) and AOSD technologies, to represent separately application-specific functions from extra-functional agent properties. This separation improves the internal modularization of the agent architecture, which is based on components and aspects, and contributes to enhance the adaptation, reuse and maintenance of the software agent.

The Malaca agent model is used from the detailed design phase to implementation. At the detailed design stage the model is use to design the internal architecture of each agent of the system. These guidelines are helpful to design agent architectures in Malaca: (i) The application-specific agent functionality is provided by independent software components; (ii) any extra-functional property (in particular those that concerns agent communication) is encapsulated by aspects. These properties are first identified from the requirements and analysis phases, and can be platform-dependent, application-dependent, or MAS-dependent. We have separated as agent-specific aspects coordination, distribution, representation, tracing, security, and learning. The agent architecture is described in terms of the components and aspects, their composition rules and a high-level description of the agent supported interaction protocols. All the descriptions (the agent architectures and interaction protocols) are supported by MaDL and ProtDL, two XML-based agent-specific description languages defined in the context of the Malaca approach.

Fig. 1 shows a UML class diagram describing the (partial) organization of the internal architecture of the vehicle agent. The depicted components and aspects are derived from the application of the Malaca guidelines to the specification of the vehicle agent of our example described above (the gas station recommender service). The architecture of vehicle agent includes four components providing data and functions specific to VANETs: a *GPSController* component, which represents the GPS receiver and provides functions related to the vehicle route and location; the *SpeedController* component, which provides functions for monitoring and modifying the speed of the vehicle; a *PetrolSensor* component, which represents the sensor that notifies the current level of the fuel tank; and *VehicleUserInterface* component that provides a graphical interface to interact with the user.

Context-awareness, which requires functions for gathering, managing, and disseminating context information, is represented by different elements of the agent architecture. Context information (also known as *context model*) is stored in the *KnowledgeBase* component, which represents the knowledge base of the agent, as resources. The resources *PETROL_LEVEL, SPEED_MODE*, and *PETROL_STATION_PREFERENCE* maintain updated information of the current state of the fuel, speed, and user preferences respectively. An adequate OO approach to

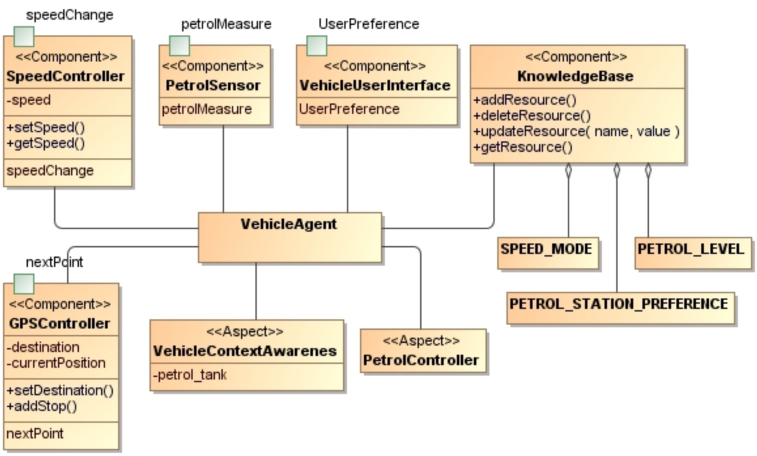


Fig. 1 Malaca architecture of the Vehicle Agent

cope with context-aware data and behavior is the subject-observer pattern (also known as Observer pattern). This behavioural pattern defines a one-to-all dependence between objects. When the state of one of the objects (the subject) changes, it notifies the change to all dependant objects (the observers).

This pattern allows varying subjects and observers independently: subjects can be reused without reusing their observers, and vice versa; and new observers can be added without modifying the subject or other observers. Regarding context-awareness, the subject is the context and the observers are the behaviours that depend on its current state. We use an aspect-oriented version of the observer pattern. There is an aspect-oriented version of this pattern, which has been demonstrated to improve the traditional solution regarding modularization [8]. Specifically, we apply the aspect-oriented observer pattern twice: Firstly it is used to characterize context acquisition and update of the context model; and secondly it is applied to characterize context-dependant behaviour. The aspect *VehicleContextAwareness* (in Fig. 1) is in charge of gathering together context information (events and messages that signify changes in the context), process it, and update the context model accordingly. This aspect is an observer of multiple subjects – speed, fuel level, and user preferences. Then, we use again the observer pattern being the subject the context model and the observers the context-dependent behaviours encapsulated in aspects. The aspect *PetrolController* encapsulates the context-aware behaviour that decides when it is required to refuel depending on the current fuel level, the speed, and the user preferences. The aspect-oriented solution helps to simplify and reuse context-awareness inside the agent.

However, the separation performed at the architectural level requires composing the aspects behaviour with the components. This process is known as *weaving*. Then the agent designer has to specify when is invoked the behaviour of the two context-aware aspects. Malaca permits the composition at specific points, called *join points*, of the agent execution flows. The Malaca join point model, which is

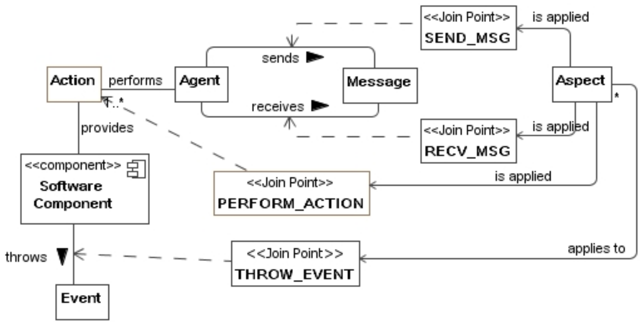


Fig. 2 Join point model in Malaca

shown in Fig. 2, defines four joint points: after the reception of an input message, before the sending of an output message, at event dispatching, and the execution of an action. Thereby, every time the agent communicates with another agent, and event is thrown, or an action invoked, the corresponding message, event or action invocation is intercepted and aspects are composed.

Weaving is guided by composition rules. The composition rules describe *point-cuts*, which are patterns that specify the set of join points being intercepted. The MaDL language allows the definition of this composition rules in XML. Each rule describes a pointcut and describes how to compose aspects when the pointcut is reached. This description can condition the composition of an aspect by specifying the values of different parameters of the join point (message fields, the name of the action and the event, value of the arguments). Note that aspect composition is not hard coded (inside the involved aspects, or any other component of the architecture). This composition mechanism is very flexible and allows defining different aspect compositions in accordance to diverse situations.

Consequently, and given that the *VehicleContextAwareness* aspect gathers information to update the context model, its behavior should be invoked every time: (i) the *PetrolSensor* component throws an event with the current tank level; (ii) the *SpeedController* dispatches an *speedChange* event*;* or (iii) the user updates his/her preferences. On the other hand, the *PetrolController* aspect should be applied every time any of resources of the context model (*PetrolLevel*, *SpeedMode* or *Petrol_Station_Preference*) is updated (method *updateResource*). The composition rules (in XML) are shown next:

```
<WeavingRule JoinPoint="THROW_EVENT">
 <OrCondition>
     <Property name="PetrolMeasure"> <Property name="SpeedChange">
     <Property name="UserPreference">  </OrCOndition>
 <ApplyAspect id="VehicleContextAwareness" relevance ="CRITIC"/> </WeavingRule>
<WeavingRule JoinPoint="PERFORM_ACTION" name="UpdateResource"
target="KnowledgeBase">
<OrCondition>
  <Property param="name" value="PetrolLevel"/> <Property param="name"
value="Speed_Mode"/> <Property param="name" value="Petrol_Station_Preference"/>
</OrCondition>
<ApplyAspect ="PetrolController" relevance ="CRITIC"/> </WeavingRule>
```

4.2 Implementation of the Context-Aware Aspects

This section briefly describes the implementation of the two context-aware aspects. These given implementations are very simple, but a more complex and appropriate context modelling and decision techniques for this application domain could be used. The Malaca agent framework provides a base class named *Aspect* for developing aspects. This class defines a common set of attributes and methods, which facilitates the identification and composition of aspects. Four methods encapsulate the aspect behaviour for each join point (*handleInputMessage*, *handleOutputMessage*, *handleEvent* y *handleAction*). The context model is represented by an ontology that describes the terms and set of values of the variables that define the context (i.e. the values of the resource PETROL_LEVEL are RESERVE, LOW, LOWER_HALF, MEDIUM, UPPER_HALF, HIGH, and FULL).

The behaviour of the *VANETControlAwareness* aspect is implemented in the *handleEvent()* method, which includes the event intercepted. Currently, the implementation of this method delegates event processing to a particular method that processes the contextual data to update the context accordingly (*updatePetrolLevel-SpeedMode-PetrolStationPreference*). This manner, if the event notifies a new fuel measure, it is invoked the *updatePetrolLevel()* method, which classifies the measure and updates the resource PETROL_LEVEL if required. Following, it is given part of the implementation.

```
public class VehicleContextAwareness extends Aspect {
              public Object handleEvent(Object event)
              {… if (_event.getName().equals("PetrolMeasure"))
                  {    updatePetrolLevel((Integer)_event.getValue());      }
                …  }//end of handleEvent();
public void updatePetrolLevel(Integer current_petrol)
{ …;updateResource("PETROL_LEVEL",VANETContext.UPPER_HALF);…}}
```

The behaviour of the *PetrolController* aspect is codified in two different methods: *handleAction* and *handleMessage*. The former is invoked when the context changes and invokes a specific function to manage it accordingly; and the latter when the agent receives an ACL message from a service agent in the proximity. Given a new context, the aspect decides if it is required to refuel. If refuel is required then current user preferences are consulted to find a preferred nearby gas station. The aspect maintains a list of close and preferred gas stations, which is updated with notifications of service agents. Such ACL messages are intercepted and processed in the *handleMessage* method. Finally, the aspect notifies the user the situation. Part of the implementation of this method is shown next.

```
public class PetrolController extends Aspect {
   RoadNetworkService _nearest_gas_station = null;
public boolean handleAction(Object action)
   {  …        if(_context_var.equals("PETROL_LEVEL"))
        { … considerPetrolLevelChange(_new_petrol_level);…}}
public Object handleInputMessage(Object message) {
 ACLMessage _inform = (ACLMessage)message;
 updatePreferredGasStation(((RoadNetworkService)_inform.getContent()));}
public void considerPetrolLevelChange(String _petrol_level)
{  if((isLow(_petrol_level)&isFast(_speed_mode))
    |(isReserve(_petrol_level)) {
        getSomePetrol();  }}
```

5 Conclusions and Future Work

This work presents the use of context-aware Malaca agents for AmI applications. The aspect-oriented approach of Malaca contributes to facilitate the development of the context-awareness property in software agents. Context-awareness concerns (contextual data gathering and context-aware adaptation) are modelled as aspects, enhancing their reuse and evolution. Context information is accessed in a non-intrusive mode for the context components, avoiding undesirable dependencies between the context model and the context-dependent behaviours inside the agent. In [8] we demonstrate that the Malaca model achieves better separation of concerns (or agent properties as context awareness), so the internal agent components are not coupled and only encapsulate a single concern (i.e. they are cohesive). In order to show this a metric suite was applied to compare an aspect-oriented approach as Malaca as opposed to non aspect-oriented ones (for example those using the JADE platform).

We are currently improving the current implementation of Malaca agents in order to reduce computational cost and response time for their execution in AmI environments. In addition, a vehicle simulator, for the practical experimentation of agents for VANETs, is currently under construction.

Acknowledgements. This work has been supported by the Spanish Ministry Project RAP TIN2008-01942 and the regional project FamWare P09-TIC-5231.

References

1. Weiser, M.: The computer for the Twenty-First Century. Scientific American 165 (1991)
2. Henricksena, K., Indulskab, J.: Developing context-aware pervasive computing applications: Models and approach. Pervasive and Mobile Computing 2, 37–64 (2006)
3. Yin, M., Griss, M.: SCATEAgent: Context-Aware Software Agents for Multi-Modal Travel. In: AAMAS Workshop on Agents in Traffic and Transportation (2004)
4. Chen, H., et al.: Intelligent Agents meet the Sermatinc Web in Smart Spaces. IEEE Internet Computing, 2–12 (November-October 2004)
5. Sánchez-Pi, N., Carbó, J., Molina, J.M.: JADE/LEAP Agents in an AmIDomain. In: Corchado, E., Abraham, A., Pedrycz, W. (eds.) HAIS 2008. LNCS (LNAI), vol. 5271, pp. 62–69. Springer, Heidelberg (2008)
6. Dantas, F., Batista, T., Cacho, N.: Towards Aspect-Oriented Programming for Context-Aware Systems: A Comparative Study. In: Proc. of the SEPCASE at ICSE 2007 (2007)
7. Amor, M., Fuentes, L., Troya, J.M.: Training Compositional Agents in Negotiation Protocols Using Ontologies. J. Integrated Computer-Aided Engineering 11(2) (2004)
8. Amor, M., Fuentes, L.: Malaca: A component and aspect-oriented agent architecture. Information and Software Technology 51(6), 1052–1065 (2009)
9. Fuentes, L., Gámez, N., Sánchez, P.: Aspect-Oriented Executable UML Models for Context-Aware Pervasive Applications. In: MOMPES 2008, pp. 34–43 (2008)
10. Munnelly, J., Fritsch, S., Clarke, S.: An Aspect-Oriented Approach to the Modularisation of Context. In: PerCom 2007. IEEE, Los Alamitos (2007)

Reasoning on a Semantic Web Based Context-Awareness Middleware*

Alberto García-Sola, Teresa García-Valverde, and Juan A. Botía

Abstract. We present in this paper the OCP middleware for context information management. More specifically, we focus on the reasoning aspects of the middleware. In order to provide the OCP middleware with reasoning capabilities, we have integrated a semantic web ontology management API based on Jena and Pellet, developed in our lab, i.e. the ORE API. We present some projects in which the reasoning capabilities have been applied and some examples trying to show the power of this open source implementation.

1 Introduction

Recent and growing development of context awareness systems in the last years has made us require of systems capable to manage this context so that we can focus on manipulating this information. Our proposal is OCP, a middleware for managing context information with the following characteristics.

Event oriented. In OCP, there are two roles, producers and consumers of context. Context producers can send information at any time, so that context consumers interested in this information will immediately receive it. Consumers can set their own interests to receive context updates of some concrete entities that interest them, or even all instances of an entity type.

Alberto García-Sola · Teresa García-Valverde · Juan A. Botía
University of Murcia
e-mail: {agarciasola,mtgarcia,juanbot}@um.es

* This research work is supported by the Spanish Ministry of Science and Innovation under the grant AP2007-04269 of the FPU program, the Séneca Grants program through the 09493/FPI/08 grant and in the scope of the Research Projects TSI-020302-2009-43, TIN2008-06441-C02-02, through the Fundación Séneca within the Program "Generación del Conocimiento Científico de Excelencia" (04552/GERM/06).

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 147–155.
springerlink.com

Distributed. OCP offers a client/server architecture. The client only needs a few lines of code to communicate with the server and handle all context. OCP server is responsible for accomplishing reasoning and distribution work for the clients, allowing interaction with lightweight devices (i.e. sensors and actuators).

Ontology oriented. Context information is organized and stored using OWL ontologies. Ontologies are a powerful tool for knowledge representation, and there are several authors who describe them as the most promising tool to represent context information [8]. Two ontologies are used in OCP, context ontology and domain ontology. Domain ontology includes every element (either structural or instance) of the domain that we are working in. The context ontology gives us some useful information for OCP to manage context. Furthermore, OCP is extensible with already defined ontologies, e.g. SOUPA, CONON, ULCO, etc.

Context history. The context history is a powerful and flexible OCP tool that stores a historical of every context information (or what you specify). This allows us to perform temporal reasoning and learning through adaptation as well as perform data mining using historical data through the API that offers OCP. Depending on the system, it is possible we do not want to store all information. For this, OCP offers different storage modes: ***Store nothing***; ***Store everything***; ***Store only new events***. Thus, OCP would only store events involving a change of context. If we have a sensor that every certain time tells us the temperature, you may be interested in only keeping temperature changes. ***Store intervals.*** Unlike the former, we store intervals.

Context reasoning. OCP includes reasoning, endowing the system with the capabilities of inferring new information. This way OCP clients stick to send and receive information in a way that they understand, letting OCP the process to give meaning to that information.

Thanks to OCP we can, with a few lines of code have full control over the system context, and even infer not explicit new information. In this paper we focus on the last part, the context reasoning. In the next section (2) the OCP context reasoning module is described deeply, including some examples (subsection 2.1). In section 3 we discuss about other works related with context reasoning to end up with conclusions and future work (section 4).

2 Context Reasoning within OCP

OCP is a middleware that allows us to control the context in a simple and distributed way between producers and consumers. The use of semantic context information based on a predefined domain ontology gives us great expressiveness and versatility. In general, the producers' logic is limited to generate data, and send it to the middleware regardless of the ontology, being the middleware responsible for providing it a semantic meaning within the domain. However, it is possible that this data in that format is not relevant to consumers, needing reasoning on it to make explicit certain implicit information. OCP performs this reasoning, thus freeing both consumers and producers of the need to include more complex logic and gather information irrelevant to them. Thus, domain logic (i.e. any relevant information for the applications about the entities in the environment and their relations) is centralized in a single point (i.e. OCP). Take, for example a temperature sensor. The sensor just generates temperature data. Depending on the domain, a temperature of 36° C may mean different things. If we are measuring a patient temperature, it may mean that the temperature is correct, if we are measuring a room temperature it may be too high and we should turn the air conditioner on. However, the sensor is unaware of this. OCP is the one who gives meaning to this information using reasoning.

In Semantic Web based representations, part of this new information generated by reasoning can always be obtained by the so-called "axiomatic reasoning", using certain implicit and universal rules for all ontologies. For instance, if we define the relation *equalsTo* as symmetric, when we add the relation *class1 equalsTo class2*, we implicitly know that *class2 equalsTo class1*. However, there is some domain information that must be described in the form of rules. These rules are needed to provide this new information, since they are domain dependent, and, possibly, dynamic (may change over time). OCP supports two types of reasoning: to detect and correct inconsistent context information and to derive higher level context information. The latter type of reasoning is based on properties like symmetry (as aforementioned) and transitivity and user-defined rules. We will focus on the latter type of reasoning, intended to derive new information.

For this reasoning we use ORE-API[1]. ORE-API is a set of tools developed in our lab for reasoning using Semantic Web used by ORE (Ontology Rule Editor), graphical tool used for easy ontological rule sets creation, debugging and validation. It is a powerful tool that allows read, modify and create new rules as well as reason information using different methods. We chose to use an API instead of reasoning tools such as Pellet directly. This approach has advantages and disadvantages. On one hand, ORE-API offers various types of reasoning (combinations of different reasoners), possibility to modify the rules easily, great versatility, updates and has been tested in different environments. Furthermore, using ORE-GUI we can create/view/modify graphically the different rules used in OCP (shown in figure 3 editing a rule). ORE-API also offers a powerful and easy programming interface to read, create and modify rules in different formats and export capabilities. All reasoning logic is encapsulated, so that any change is made simple. However, the election of ORE-API also has some drawbacks, since it creates an extra layer of abstraction with a possible performance penalty.

[1] http://ore.sourceforge.net/

When OCP does not use the reasoner, the operation flow when new context is notified by a producer is the following: (1) Receive and manipulate the message. (2) Insert/modify ontology information. (3) Update the history. (4) Notify concerned consumer. Reasoning entail the possible generation of new knowledge, to be updated itself into the historic and possibly notified to different consumers. Since new information is needed to be in the ontology to reason with it, the only temporal constraint is the reasoner to be called after the insertion/modification of information in the ontology. In our case, we have decided to make this reasoning after the first notification to consumers for several reasons. On the one hand, we avoid unnecessary waiting for the consumers to be notified, probably not interested in the new created context. On the other hand, this way there is some sequence (also in history) about events, giving priority to information received prior to that generated, avoiding cases of consumers who interpret the generated information before producing potential inconsistencies. Note that the new inferred information should be introduced in OCP so that it can notify potential consumers and generate the historical of this information, which will be treated as information generated by a producer for all purposes.

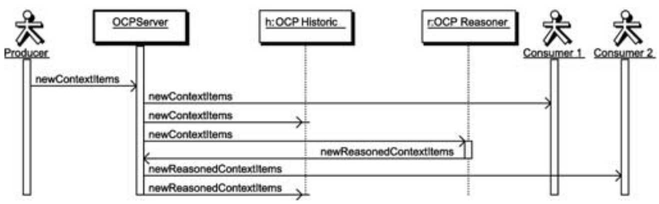


Fig. 1 New context arrived to OCP. Sequence diagram.

In figure 1 we can see a sequence diagram of how OCP handles a new context event. The first thing done by OCP when receiving new context information (*newContextItems*) is notifying the subscribed consumers, as discussed previously. Then, reasoning is performed on the new information received, and if any new knowledge is inferred (*newReasonedContextItems* in the figure), it is sent to consumers (if any subscribed to that new information).

Reasoning in OCP is compound of two parts, the axiomatic reasoning and the rule-based reasoning. From the range of options offered by ORE-API, we have used Jena (Rule-based Reasoning Engine) + Pellet (as Ontology Reasoning Engine) reasoning by default. On the one hand, there is the Ontology Reasoning, carried out by Pellet. Pellet is an open-source Java OWL DL reasoner based on the tableaux algorithm. Other approaches such as Bossam (translate OWL-DL into rules and give the rules to a forward chaining engine) do not fully cover OWL-DL (cannot cover the full expressivity of OWL-DL due to many incompatibility between Description Logic and Horn Rule formalisms). Moreover, Pellet integration with Jena (ontology management is done in OCP using Jena) and ORE-API is straightforward. Knowledge models based on DL ontologies are usually divided into TBox (terminological) and ABox (assertional) components [4]. The TBox contains the vocabulary and schema that define domain concepts, their properties, and the relationships (called roles in DL) among them. Therefore, once OCP has started, TBox never change, since the schema remains unchanged, and reasoning over TBox must only be done once. However, ABox, which requires a previous TBox reasoning does change through time, hence, reasoning to ABox must be done every time a new item arrives the system.

On the other hand, there is the Rule-based Reasoning, carried out by Jena. The selected rule reasoners should provide the knowledge inferred by the rules isolated from the rest of the knowledge inferred by other reasoning processes. This feature is required by OCP to be aware of the new information and let it process properly, as well as for debugging tasks. The Jena rule reasoner makes this distinction and split the inferred knowledge from the base knowledge whereas the Pellet rule reasoner does not. Jena Rule-based Reasoning has diverse built-in engines, which support forward chaining, tabled backward chaining and hybrid execution strategies [3]. Rules have been introduced in the ontology using the Semantic Web Rule Language (SWRL)[2] based on a combination of the OWL-DL and OWL-Lite sublanguages of the OWL with the Unary/Binary Datalog RuleML sublanguages of the Rule Markup Language. These rules has a semantic equivalence to well researched description logics. SWRL allows us a great power and expressiveness. Its ability to write our own built-ins allows us to model new functionality to interact with the history and perform temporal reasoning in a more natural way.

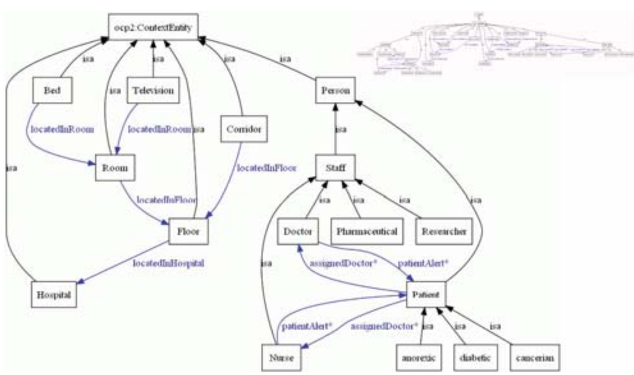


Fig. 2 Domain Ontology

[2] http://www.w3.org/Submission/SWRL/

2.1 AmI Example

To better explain the role of context reasoning in context-aware computing, we present a real AmI scenario applied to e-care (health care), described by the ontology of a simplified hospital shown in figure 2, used in the project CARDINEA[3]. In the same figure (2) we see (in miniature) the same ontology as it would be when running inside OCP, including the historic. As we see, all entities inherit from *ContextEntity*. By defining this hospital example we will show some rules and how they would behave in OCP.

The scenario is based on a hospital, where there are different kind of patients and the staff, where the doctors are included. For the rules we are going to show, we use an actuator and two different sensors. The actuator is a kind of bleeper which sends alerts to the doctors when there is an emergency, and in OCP is represented as a consumer of context (i.e. is subscribed to some part of context and notified when it changes). One of the sensors measures the glucose level of patients, and the other is just an emergency button placed in a concrete bed, which sends that there is an alert in a concrete bed. Both sensors are context producers in OCP. The ontology is populated with a specific kind of patient instance *diabetic* (id: diabetic_1), which is *locatedIn* a *bed* (id: bed_2) and has an *assignedDoctor* (id: doctor_3).

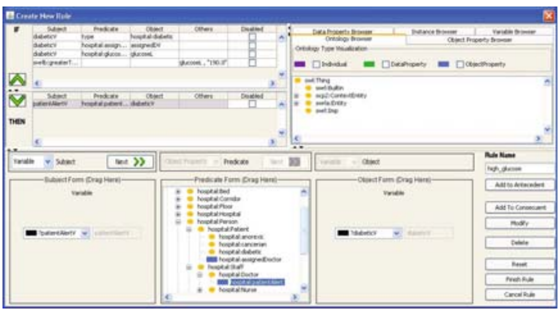


Fig. 3 Example's rule edition using ORE-GUI

In figure 3 there is a first basic rule for a diabetic, shown using the ORE-GUI editor. What the rule basically does is to send the doctor an alert through the bleeper when the glucose level of a diabetic is above a certain value. In this concrete rule there is no need to use the axiomatic reasoner or even an ontology, since the rules use the information directly sent from the sensors, and a simple rule-based reasoner could do the same. So, in this example, if the glucose_level sensor connected to the diabetic patient sends to OCP a high value, OCP would get it and start the reason process. In the given example this would be the reasoned information (only rule-based information showed,

[3] http://cardinea.grupogesfor.com/

not the axiomatic reasoned information):
[hospital#doctor_3, hospital#patientAlert, hospital#diabetic_1]
This triple means that *doctor_3* sets the object property *patientAlert* value to *diabetic_1*. Then, OCP gets this new information and sends it to the bleeper (context consumer of *patientAlert* for *doctor_3*). Therefore, an alert is sent to *doctor_3* about the patient *diabetic_1*.

Bed(?swrl:bedV) $\wedge$ *manual_alarm(?swrl:bedV, ?swrl:manualAlarmV)* $\wedge$ *swrlb:equal(true, ?swrl:manualAlarmV)* $\wedge$ *contains(?swrl:bedV, ?swrl:patient)* $\wedge$ *assignedDoctor(?swrl:patient, ?swrl:assignedDoctorV)* $\wedge$ *Patient(?swrl:patient)* $\Rightarrow$ *patientAlert(?swrl:assignedDoctorV, ?swrl:patient)*

The rule described above is a different and more generic rule valid for every *patient*, and, therefore, for this *diabetic* in the example, since it is a subclass of *patient*. The rule takes the event sent from the emergency button placed on the bed, and using that information, alerts the doctor. It checks which patient is on the bed using the *contains* object property. However, in the ontology there is only information about where are the patients located (*locatedIn*). *Contains* is the inverse of *locatedIn*, thus, using the axiomatic reasoner the system can obtain that information. The rest of the rule finds which doctor is assigned to that concrete patient using *assignedDoctor* property and alert him trough the bleeper (context consumer) telling which patient is in alert.

All in all, from just a simple event (button-pressed or glucose-level) we can derive domain specific high-level information using explicit rules and the reasoning engine available in OCP, keeping the actuators and sensors logic extremely simple and leaving information manipulating complexity to OCP.

3 Related Work

First OCP version dates back to 2006. Since then to the current OCP version (OCP2) system has evolved and improved in many aspects (e.g. ease of use, distributed, history, reasoning...). There are several platforms and middlewares for context management with diverse approaches. In [8, 6] we can find a detailed survey of some of them, including different approaches (e.g. object-oriented models, ontology based models, key-valued models...), to conclude that ontologies are the more promising alternative.

Regarding the use of reasoning in these systems, there is also a wide range of works. In [2] we can find a survey discussing many of them. A variety of context reasoning schemes have been proposed to deal with context reasoning, including Bayesian networks [7], Dempster-Shafer Theory [9], logic-based [1] and ontology-based context reasoning schemes among others, but only ontology-based reasoning schemes incorporate semantics into context representation and reasoning. Evolving from semantic networks, ontology provides a structural model to handle complex and disparate information.

4 Conclusions and Future Work

In this paper we have presented the OCP middleware for context information management. We have shown in detail how reasoning is performed and what advantages and drawbacks offers, and a few examples applied to AmI to better understand it. Documentation, source code and latest OCP version is available at the web[4]. OCP has been used in various research projects and is currently deployed in several real systems in operation, among which we highlight the European project POPEYE (IST-2006-034241), CARDEA (FIT-350101-2007-14) and CARDINEA (TSI-020302-2009-43), so we can ensure that systems context-aware are no longer merely a research tool. However, there are still many open possible improvement in which we are already working which offer us a way to create more advanced services.

Firstly, the history is offering very promising results related with temporal reasoning. Furthermore, using collaborative filtering algorithms on ontologies as [5] can provide the system with recommendation capabilities. Another interesting task is the total distribution on different servers. Using a fully distributed structure we can use small storage devices and release the CPU load when reasoning In addition, it would provide a more robust system in ubiquitous environments where the network may not always be available. Finally, we are working with some techniques to improve the efficiency.

References

1. Alsinet, T., Chesñevar, C.I., Godo, L., Sandri, S., Simari, G.: Formalizing argumentative reasoning in a possibilistic logic programming setting with fuzzy unification. International Journal of Approximate Reasoning 48(3), 711–729 (2008)
2. Bettini, C., Brdiczka, O., Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A., Riboni, D.: A survey of context modelling and reasoning techniques. Pervasive and Mobile Computing (2009)
3. Carroll, J.J., Dickinson, I.: Jena: implementing the semantic web recommendations. In: Proceedings of the 13th international World Wide Web conference, pp. 74–83. ACM, New York (2004)
4. De Giacomo, G., Lenzerini, M.: TBox and ABox reasoning in expressive description logics. In: Principles of Knowledge representation and reasoning, pp. 316–327. Morgan Kaufmann, San Francisco (1996)
5. García-Sola, A., Botia, J.A.: Creation of Semantic Overlay Networks Based on Personal Information. In: Omatu, S., Rocha, M.P., Bravo, J., Fernández, F., Corchado, E., Bustillo, A., Corchado, J.M. (eds.) IWANN 2009, Part II. LNCS, vol. 5518, pp. 75–82. Springer, Heidelberg (2009)
6. Hong, J., Suh, E., Kim, S.J.: Context-aware systems: A literature review and classification. Expert Systems with Applications 36(4), 8509–8522

[4] http://darwin.inf.um.es/ocp/dokuwiki/

7. Ma, Y., Kalashnikov, D.V., Mehrotra, S.: Toward managing uncertain spatial information for situational awareness applications. IEEE Transactions on Knowledge and Data Engineering, 1408–1423 (2008)
8. Linnhoff-Popien, C., Strang, T.: A context modeling survey. In: Workshop on Advanced Context Modelling, Reasoning and Management UbiComp. Citeseer (2004)
9. Zhang, D., Cao, J., Zhou, J.: Extended Dempster-Shafer Theory in Context Reasoning for Ubiquitous Computing Environments. In: Proceedings of the 7th IEEE International Conference on Embedded and Ubiquitous Computing (2009)

Agent-Based AmI System Case Study: The Easy Line + Project

Armando Roy Delgado, Rubén Blasco, Álvaro Marco, Diego Cirujano,
Roberto Casas, Armando Roy Yarza, Vic Grout, and Richard Picking

Abstract. Smart environments include context-aware computing to enhance system capabilities which affect user interaction positively. A context-aware application should collect contextual information through different ways depending on technology availability and handle this information adequately to execute actions or warnings, present information or modify the environment. Ambient intelligence (AmI) habitually means system complexity and heterogeneity. This paper describes an AmI system case-study, developed for the Easy Line + project, a decentralized intelligence carefully structured in agents grouped by roles in domains to improve system package interaction and integration. The main purposes for this implementation are: avoid complexity and inconsistence between the elements of the system and increase system´s performance. The Easy Line + project is primarily oriented to monitor and control white goods and to assist in emergency situations though it is deployed all over the home environment and can be extended, more easily thanks to the multi-agent approach, to control and monitor other devices or services at home.

Keyworks: Ambient intelligence, context-awareness, domain, agent.

1 Introduction

The application of information technologies to home environments is mainly characterized by the integration of networked computational devices into the physical context. The concept is also denominated with the term ubiquitous computing, which involves ubiquitous communications and user interfaces. AmI implies the

---

Armando Roy Delgado · Rubén Blasco · Álvaro Marco · Diego Cirujano · Roberto Casas · Armando Roy Yarza
Grupo Tecnodiscap, Universidad de Zaragoza, Spain
email: {aroy,rblasco,amarco,ciruman,rcasas,armanroy}@unizar.es

Vic Grout · Richard Picking
Centre for Applied Internet Research (CAIR), Glyndwr University, North Wales
email: {v.grout,r.picking}@glyndwr.ac.uk

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 157–164.
springerlink.com

existence of another abstraction level on top meaning adaptation, personalization, unobtrusiveness and anticipation—expressions that commonly cope with other AmI projects [1][2][3] also oriented to social inclusion—.

Context-sensitive systems must be flexible enough to perform according to the requirements independently from the architecture, location, user profile and the technologies involved. Furthermore they should reconfigure dynamically to accommodate users' needs, taking into account a wide range of users and situations.

To develop a system with such characteristics there must be processing units distributed all over the physical context, as well as wearable devices. Ubiquitous computing enables these objects to communicate with each other and the user transferring data from one point to another dynamically as required. A wide range of devices can be used to computerize homes like cameras, location systems and a variety of sensors, whose function is to collect contextual information. Aside from data collecting devices AmI systems include a variety of human-machine interfaces (HMI) such as displays, touch screens, remote controls, TV, hand held devices and so.

Finally networking devices have to be incorporated to put everything together and some others to act upon control and coordinate everything—e.g. computes or servers—. The type and number of these devices may vary according to the place and needs of the user. Altogether, the integration of networked computational devices into the targeted environment gives the possibility to reduce the issues that nowadays arise about the risk of people losing independence at home and permits to lower the levels of social exclusion in the society that are getting higher as predicted years ago [4].

To increase people autonomy at home is one of the main objectives of the Easy Line + project [5]. Easy Line + project aims to allow elderly people to remain in their homes living independently thanks to AmI, by facilitating realization of daily tasks such as preparing foods or washing clothes, detecting risky or abnormal situations, and evaluating their quality of life in that scenario.

In this paper, we firstly make a constructive analysis about the advantages of multi-agent systems in AmI. After that, there is a generic description of the elements of the system grouped by domains, explaining the main tasks of each domain and the interactions between them, and the enumeration of the agents involved in each domain and their relationships. Finally, conclusions are presented.

2 Reasons to Apply a Muti-agent System Approach in AmI

Distributing the intelligence makes the system easier to adapt and more scalable. In other words AmI systems have to be able to add extra functionalities, perhaps overlooked in the first instance, to cope with user requirements over time.

In contrast, a centralized intelligence in AmI is predisposed to fail because it often implies risks that are not feasible for home environments where people spend most of their time. There might be a lack of processing power, memory bottlenecks or a loss of centralized data that may affect the performance of the system making it unstable or unsuitable to fulfill user needs by that moment. In addition, in a centralized system, a network failure or a faulty networked device may collectively affect the rest of the network as some researchers stated like Park et al. [6].

A Multi-agent system perspective has numerous advantages, almost certainly the most important related to AmI systems are:

- Monitoring and control versatility
- Better resource allocation
- Facilitate system design
- Allow modularity and flexibility
- Robust behavior on automated processes

Many communication protocols, network coordinators and a flexible device manager make the system more versatile and therefore more capable to influence the life of the user positively. In addition, having agents in diverse physical locations requires less processing power and memory per device minimizing bottleneck incidents. Furthermore if all agents have specific roles that can be integrated with the rest of the system, the result is a modular architecture flexible enough to accept changes in certain modules and/or to add new ones. On top of that, the system can contain some reactive agents—automatons that receive an input, processes it and produce an output—which perform better in certain situations as they are unaffected by other system processes but are still integrated with the rest of elements.

However developers have to bear in mind that the application of multi-agent systems to AmI cannot be reduced to pursue those universal benefits as user requirements tend to be more explicit. The following sections enter into more detail about how Easy Line + is going to affect the lives of the users and which multi-agent architecture is selected to adjust the system the user needs.

3 Systematization of the Easy Line + Project into Agent Domains

A multi-agent system consists of some individual agents each of them oriented to carry out specific tasks. Collectively, a set of agents are able to achieve tasks that will be rather impossible to carry out in a centralized system having only a single agent. The use of multi-agent systems has increased considerably during the past few years, nowadays is a common term constantly used in context-awareness systems and service oriented architectures [7][8][9].

In the Easy Line + project, we seek to obtain the benefits of using a multi-agent architecture explained in the previous section. In addition, we aim to accomplish some other specific goals: context data mining; reusability and management of data; user activity monitoring; non-intrusive system responses; learn user capabilities; content adaptation [10][11]; user profile identification; and accessible; personalized and coordinated HMI.

To complete these tasks, the system has to incorporate many devices and employ different network technologies; it also requires an assorted set of software/hardware specifications/implementations and to customize or tailor the information according to the user's preferences [11].

For that reason it is significant to classify agents in different groups or domains according to their roles and characteristics to increase specialization and achieve a better performance. Besides, the interoperability of all the domains has to be consistent and fluent to minimize system latency. We have identified five domains in the Easy Line + project as shown in figure 1.

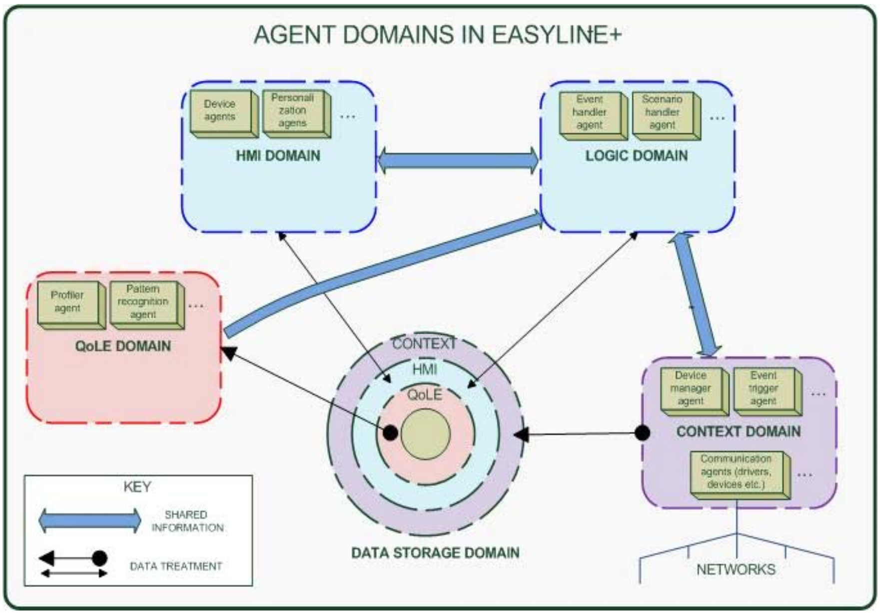


Fig. 1 Agent Domains of the Easy Line + project

- **HMI domain** will be the one with closer contact with the user. It includes all agents related to the interfaces: physical devices, personalization of interfaces, human-machine interaction accessibility and HMI interface coordination/monitoring.
- **Context Management domain.** Information about the context is important, the more the system knows the better will acclimatize the environment to the user. We include RFId information, environmental sensors, and the status of appliances, each white good must be able to inform about its status and also be commanded remotely. RFId is integrated in the system to recognize food, garments and medicines in the environment (informing about food about expire, recommended program for the laundry, etc.). Each technology needs to be integrated involving a set of agents such as driver agents, communication agents, context managing agents, networked devices (not HMI) etc.
- **Logic domain.** There must be a set of agents controlling system´s logic. In a way, this domain could be the glue which put the other domains together although bearing in mind this is not entirely strict. Logic agents are in charge of

understanding predefined situations (or scenarios) and support the rest of the system to act upon them.

- **Data Storage domain** copes with distribution, classification and storage of data among the databases.
- **Quality of life evaluation domain (QoLE)** is mainly oriented to evaluate user´s behaviour, it involves neural networks, professionals to evaluate the results of the neural networks, and system configurations according to the patterns encountered in the neural networks.

4 Agent Specifications in the Easy Line + Project

Once domains are sorted out and their roles established, each domain is broken down into tasks and services. This part is critical in order to define the agents of each domain. Each definition must include the processes each agent is capable of, which data is going to be handled and to which other agents is going to be forwarded to. In the end the resulted modularity has to be consistent and smooth, in other words, agents do not have to wait long for other agents to finish their tasks or do not have to be affected much about the errors other agents could have. Moreover the agents have to be adjustable and/or extendible in case other agents are introduced in the system and/or specifications are re-defined.

The system, as it is at the moment, has several definite agents most of them predefined according to the system requirements established initially. The resulting hybrid architecture tries to combine deliberative and reactive aspects by

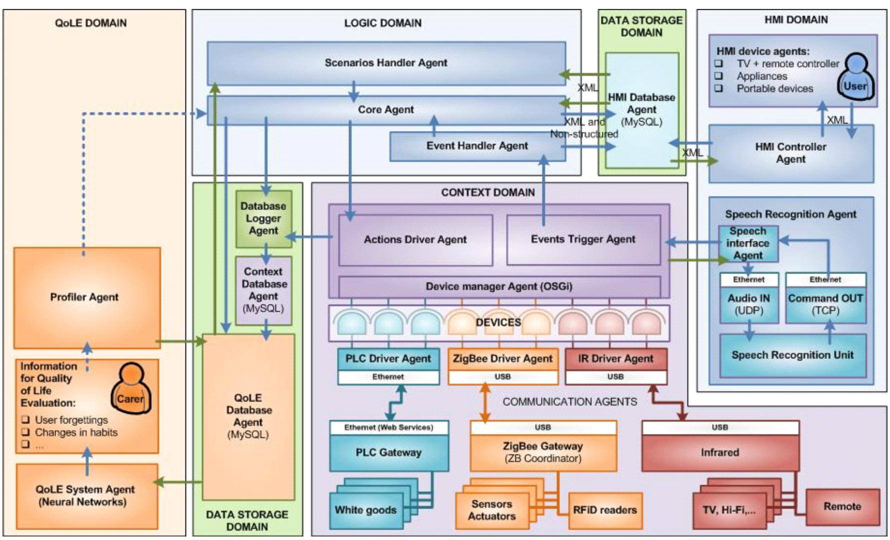


Fig. 2 Agents and their relationships in Easy Line +

combining deliberative and reactive agents [12]. This architecture of agents is a good orientation to developers to find the proper programming model which, as Ricci et al. said [13], is a key point to build a complex and modular system.

As there are too many agent specifications to make an extensive description of them, we will enumerate and describe briefly some of them, emphasizing their key characteristics and relationships with the other agents in the system (see figure 2).

- **HMI Device agents.** Several types of clients can be used to manage the system: mobile devices such as PDA, smart phone, wearable device, ultra-mobile PC, touch screen device etc; and fixed devices such as computers, digital TV and fixed screens that will be used as a centralized control.
- **HMI Controller agents** integrate each HMI device with the system using a bi-directional communication with the device and the HMI part of the database. It does not connect directly with the logic domain agents, while this communication goes through the database and therefore remains registered to keep track of timestamps and any data handled in all user interactions through any HMI.
- **Actions Driver agent** processes data from the core agent regarding user interactions or automated controls, and forwards actions to the device manager agent which will inform the driver about which tasks have to be carried out in the network(s). It also keeps track of the changes by telling the database logger to store environmental information in the context database.
- **Events Trigger agent** is in charge of inform the logic domain about relevant changes that take place in the environment. Afterwards the logic domain will have to decide which ones have to be notified to the user and how.
- **Device manager agent** creates a semantic representation of the devices: appliances, sensors, etc. This representation is done using the OSGi standard. This approach will permit other implementations to include these devices and make use of their features easily. OSGi framework is a good solution to manage the information provided for the devices. It makes easy to keep separate the Hardware issues and requirements of the storage of data provided for them.
- **Driver agents** are devoted to gateway control, device registration and device publication. Published devices will be accessible by the logic domain an consequently by the users. There are three driver agents developed at the moment to fulfill the specific needs of the EasyLine+ system: PLC, ZigBee and Infrared drivers.
- **The Core Agent** will communicate with HMI and parse a predefined XML message file. This module will be responsible to send the correct information to the HMI controller agent according to the user profile and choices. Moreover it will have to save user data into the HMI database and the QoLE database agents, this last one to be useful to the neural network agents.
- **Scenarios handler agent** analyzes the HMI database agent (every few milliseconds) looking for situations or scenarios that require intervention from the system. When a situation requires direct user interaction, for example when the fridge's door is open for more than a given time, when the washing machine program is inappropriate for the garments in, etc, then it informs the core agent which will communicate with the HMI Controller agent. These situations are

"fixed rules", which means that it is known what the system will do in a given (identified) context.

- **Event handler agent.** Any significant changes in appliances or sensors that the Context Manager writes into the data base are sent, in real time, to the Event handler agent. Then the core agent decides if the user must be informed through the HMI domain or not.
- **The QoLE System agent** analyzes logged data from the context looking for patterns that might indicate relevant behavioral changes. For example, more frequent oversights, capacities diminishing, etc. This is done with neural networks, and the rules are dynamically modified by the learning capability of the neural network algorithms.
- **Information for QoLE (caretaker).** Since system adaptation to user´s behavior cannot be left to neural networks only, a professional has to consider if the results are realistic. For example, there may be decisions such as: some people are absent–minded and usually forget about closing the fridge's door; this doesn't indicate low cognitive level, but if a person that never forgets about it, starts forgetting closing the door, might start having cognitive problems.
- **Profiler agent.** Users have complete profiles constructed with their capabilities (cognitive, audible, visual, user level with new technologies, etc.). If there are recognizable changes in the user´s profile this agent will update the profile and inform the logic domain about it.

5 Conclusions and Future Work

The simplification of the system due to the applied multi-agent architecture proved to be valuable. Nevertheless in Easy Line + the added value of applying user profiles to adjust the system and the inclusion of neural networks to identify human behavior patterns involves a higher complexity, a drawback that involves more risks that has to be treated carefully to avoid intrusiveness and dangerous situations.

Thanks to the agent classification by domains, all system elements could be finely tuned according to their specific functionalities and requirements without worrying too much about the influence on other elements. In addition, having distributed databases, processing power and memory greatly reduced the apparition of bottlenecks allowing the system to perform tasks parallel in different domains; making it less suitable to un-expected overloads.

To sum up, although the inclusion of several heterogeneous technologies makes the system harder to develop, it is clear that the modularization of the system into agents leads to the successful realization of the project.

Acknowledgments. This work was partially supported by the EU under MonAMI (IST-5-035147) and Easy Line + (IST-045515) projects and by the Spanish Ministry of Science and Technology under the AmbienNET Project (TIN2006-15617-C03-02).

References

[1] Emiliani, P.L., Stephanidis, C.: Universal access to ambient intelligence environments: Opportunities and challenges for people with disabilities. IBM Systems Journal 44(3), 605–620 (2005)

[2] Gonçalves, D.: Ubiquitous Computing and AI towards an Inclusive Society. In: EC/NSF Workshop on Universal Accessibility of Ubiquitous Computing: Providing for the Elderly. Alcácer do Sal, Portugal (2001)

[3] Abascal, J.: Ambient Intelligence for people with disabilities and elderly people. In: SIGCHI Workshop on Ambient Intelligence for Scientific Discovery (AISD), Vienna (2004)

[4] Aichholzer, G.: Report on changes and trends in service delivery. IST PRISMA project (IST-1999-29088) Deliverable D2.1 (2001)

[5] EASY LINE +: Low Cost Advanced White Goods for a Longer Independent Life of Elderly People (2007-2010), http://www.easylineplus.com/

[6] Park, S.B., Lee, W.H., Park, B.Y., Ahn, S.J., Chung, J.W.: An Agent System for Recovering Disabilities in the Ubiquitous Environment. International Journal of Computer Science and Network Security 7(8), 128–132 (2007)

[7] Velasco, J.R., Marsá-Maestre, I., Navarro, A., Lopez, M.A., Vicente, A.J., De La Hoz, E., Paricio, A., Machuca, M.: Location-aware services and interfaces in smart homes using multiagent systems. In: Proc. 2005 Int. Conferecence on Pervasive Systems and Computing. PSC 2005, Las Vegas, Nevada, USA, June 27-30, pp. 104–110 (2005), ISBN 1-932415-52-1

[8] Neelima, A.V., Sunitha, B.R., Aghila, C.G.: A Multi Agent, Service Reassembling Architecture for Context-Aware Systems. International Journal of Recent Trends in Engineering 1(1) (May 2009)

[9] Gloss, L., Mason, B., McDowall, J.: Multi-agent System Service and Ontology Design Methods. In: 10th IEEE Conference on E-Commerce Technology and the Fifth IEEE Conference on Enterprise Computing, E-Commerce and E-Services, pp. 453–456 (2008) (cecandeee)

[10] Moraïtis, P., Petraki, E., Spanoudakis, N.I.: Engineering JADE Agents with the Gaia Methodology, Dep. of Computer Science, University of Cyprus, pp. 77–91. Springer, Berlin (2003)

[11] Berhe, G., Brunie, L., Pierson, J.M.: Modeling Service-Based Multimedia Content Adaptation in Pervasive Computing. In: ACM CF 2004, Italy, April 2004, pp. 60–69 (2004), ISBN:1-58113-741-9

[12] Carrascosa, C., Bajo, J., Julian, V., Corchado, J.M., Botti, V.: Hybrid multi-agent architecture as a real-time problem-solving model. Expert Systems with Applications 34(1), 2–17 (2008)

[13] Ricci, A., Buda, C., Zaghini, N.: An agent-oriented programming model for SOA & web services. In: Proc. of 5th IEEE International Conference on Industrial Informatics, INDIN 2007, pp. 1059–1064 (2007)

Multiagent System for Detecting Passive Students in Problem-Based Learning

Alexandre Ádames Alves Pontes, Francisco Milton Mendes Neto, and Gustavo Augusto Lima de Campos

Abstract. A computer-supported collaborative learning environment can enable students in web-based distance education courses to interact with each other and with one or more facilitators to conduct group work. Problem-based learning (PBL) is a learning theory that emphasizes the use of collaboration and teamwork to solve problems. However, a problem that occurs frequently in the implementation of PBL is the presence of passive students, usually students who have difficulty working in teams to solve problems. In face to face teaching, in classes of appropriate sizes, the facilitator can easily detect the presence of students with this profile and try to correct this situation to improve the learning process. In distance learning, however, this is not a trivial task, mainly due to issues related to the geographic distribution of students and the lack of information about their levels of motivation. Therefore, this paper presents a multiagent system for detecting passive students in PBL in virtual learning environments to detect and correct this undesired situation and improve the learning process.

1 Introduction

Problem-based learning (PBL) is a pedagogical theory, based on Vygotsky's socio-cultural theory, which has been widely applied in recent years. According to [2],

Alexandre Ádames Alves Pontes
Postgraduate Program in Computer Science, Rural Federal University of the Semi-Arid, Mossoró - RN , Brazil
e-mail: alexandreadames@ufersa.edu.br

Francisco Milton Mendes Neto
Postgraduate Program in Computer Science, Rural Federal University of the Semi-Arid, Mossoró - RN , Brazil
e-mail: miltonmendes@ufersa.edu.br

Gustavo Augusto Lima de Campos
Postgraduate Program in Computer Science, State University of the Ceará, Fortaleza - CE, Brazil
e-mail: gustavo@larces.uece.br

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 165–172.
springerlink.com

PBL is a method in which students learn by solving problems. Learning is self-directed. Students work in small collaborative groups to solve problems. The teacher acts as a facilitator of the learning process instead of just passing knowledge [3].

PBL, when applied correctly, offers several benefits to the learning process, among which the following are particularly noteworthy [2]:

- Developing critical reasoning and creativity in students;
- Increasing students' problem-solving skills;
- Increasing student motivation;
- Helping students to apply the acquired knowledge in new situations.

The empirical studies in [3] show that students who learn through this approach have a greater ability to apply their knowledge in new problems and to use more effective strategies for self-learning than students who learn through traditional teaching methods.

The role of the facilitator is to guide students in this process, identifying possible deficiencies in their knowledge and skills necessary to solve the problem proposed. Thus, in this learning theory, rather than the facilitator simply transferring the knowledge to the students and then testing them through evaluations, he causes the students to apply their acquired knowledge in new situations. In this approach, students often face ill-structured problems and are motivated to discover, through investigation and research, useful solutions.

PBL emphasizes teamwork as a key requirement for the success of the learning process; in other words, collaboration is essential [8]. However, a frequent problem in the teaching-learning process is the presence of passive students, usually students who have difficulties of working in teams and collaborating to solve the problem or who are discouraged with the problem, with the process or with the environment.

In face-to-face teaching, in appropriately sized classes, the facilitator can more easily detect the presence of students with this profile and try to correct this situation to improve the learning process. However, in distance learning, the facilitator usually does not have all the information necessary to detect passive students in the environment because group members are often geographically distributed. Therefore, this paper presents an agent-based approach for detecting passive students in problem-based learning in virtual learning environments.

This paper is divided into five sections. Section 2 discusses related works. Section 3 provides an overview of multiagent systems. Section 4 describes the proposed multiagent system and the agents that are included in this approach. The last section presents our final remarks and discusses future works.

2 Related Works

Multiagent Systems (MAS) have been widely used in educational applications. This technology has been quite promising as an aid in collaborative learning environments, making these environments more proactive and autonomous. MAS can be used, for example, to assist in the implementation of a particular learning theory in a collaborative environment.

In [12], a simulated student architecture is designed to detect and avoid three situations that decrease the benefits of learning in collaboration: off-topic conversations, students with passive behavior and problems related to students' learning.

In [1], a collaborative learning environment is presented, called Cole, which focuses on social interaction among the participants from the learning process. This environment was built to support project-based learning. The learning process in the environment is accomplished with the aid of portfolios, which are described as planned and organized collections of works produced by the student(s), over a certain period of time. The portfolio reflects the profile of each student and of each teacher during the teaching-learning process. In [5] , another environment built to support project-based learning is presented.

In [4], a model for virtual learning environments is presented that employs intelligent agents to implement Vygotsky's sociocultural theory, focusing on the social aspect of interaction. The proposed model has several agents, among which we highlight the following: (i) the social agent, whose main goals are the construction of models for groups of students and the identification of groups of students that can cooperate in good conditions; (ii) the tutor agent, which evaluates the student's educational goals and recommends some type of activity; and (iii) the personal agents for assistance to the students, which monitor their activities and then inform other agents of the results of the monitoring.

As a distinctive feature of our work, we highlight the fact that our approach uses an animated interface agent with socio-affective features, i.e., when the problem detector agent identifies passive students, the animated interface agent tries to solve or minimize the problem by motivating the students to participate in activities and discussions. For this purpose, it uses facial expressions, gestures, sounds and text messages, depending on the degree of passivity detected. Moreover, unlike the others works discussed in this section, the proposed approach is based on PBL, which is a learning theory that has been proven to be effective [11, 10, 9].

3 Multiagent Systems

According to [7], agents are autonomous software entities that perceive their environment through sensors and perform actions on the environment through actuators, processing information and knowledge. A Multiagent System (MAS) consists of a set of autonomous agents that collaborate to solve a problem that is beyond the capacity of a single agent.

The structures of agents are defined basically from the pro-active and reactive concepts. Following the simple reflex agents and reflex agents with internal state are conceptualized according the definitions presented by [7] and [13].

3.1 Simple Reflex Agents

Simple reflex agents select an action based on current perception of the environment, ignoring previous perceptions. This structure of agent assumes that at any moment: (1) through sensors the agent receives information about the states of the environment, defined in a set of possible states, $\mathbf{S} = \{\mathbf{s_1}, ..., \mathbf{s_n}\}$; (2) a subsystem of perception, $\mathbf{see} : \mathbf{S} \rightarrow \mathbf{P}$, treats each state of the environment and maps it in a perception defined in a set of possible perceptions, $\mathbf{P} = \{\mathbf{p_1}, ..., \mathbf{p_m}\}$, which are representations of aspects of the states of **S** that are accessible to the agent for decision; (3) a subsystem of decision making, $\mathbf{action} : \mathbf{P} \rightarrow \mathbf{A}$, process the perception in **P** and select an action defined in the set of possible actions for the agent, $\mathbf{A} = \{\mathbf{a_1}, ..., \mathbf{a_k}\}$; (4) the agent performs the selected action in the environment through actuators.

To attain their objectives, the action selection mechanism can uses commonly occurring input/output associations information. In the case the information is represented as condition-action rules, according to [7], after *see* function processing is done on the visual input to establish a condition, this triggers an established connection in action function to select a satisfactory action. We have many such connections, some of which are learned responses and some of which are innate reflexes.

3.2 Reflex Agents with Internal State

In order to have a more rational performance, this type of agent saves an internal state address control world aspects that are not evident in the current perception. This state depends on the history of previous perceptions of the environment and is defined in a set of possible of internal current states, $\Delta = \{\delta 1, ..., \delta 1\}$. Figure 1 gives the structure of the reflex agent, showing how the current percept is combined with the old internal state to generate the updated description of the current state.

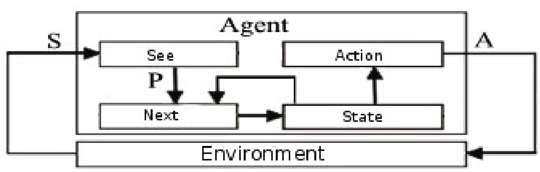


Fig. 1 A reflex agent with internal state

Therefore, there is a subsystem, *next* : $P x \Delta \rightarrow \Delta$, which maps perceptions in **P** and the current internal state in Δ to a new internal state in Δ, which will be used to select the next action. The selection of action of reactive agent with state is defined as a mapping of internal states in actions, $action : \Delta \rightarrow A$. According to [7], updating the internal state information as time goes by requires two kinds of knowledge to be encoded in the next function: (a) some information about how the world evolves independently of the agent; and (b) some information about how the agent's own actions affect the world.

4 Agent-Based Approach for Detecting Passive Students in Problem-Based Learning

Intelligent agents can perform many tasks in computer-supported collaborative learning, such as monitoring students' participation in discussions, facilitating the selection of topics for discussion, and assessing student performance in relation to the use of communication and cooperation tools available the environment, among others. The use of agents to assist with these tasks is becoming increasingly important, mainly due to the increasing number of students who interact in learning support systems, which makes it very difficult to the facilitators to manage these activities at distance.

The agent-based approach proposed in this paper is sketched in Figure 2.

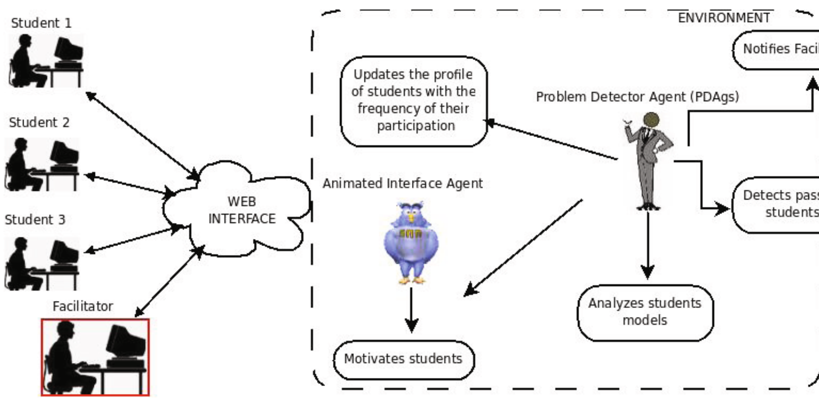


Fig. 2 Agent-based approach for detecting passive students

According to the approach shown in Figure 2, two types of agents are proposed: a Problem Detector Agent (PDAg) and an animated interface agent. The PDAg are responsible for updating the profiles of students with the frequency of their participation whenever the students use the collaborative mechanisms and for detecting passive students, by analyzing the profile of existing students. After detecting a passive student, it notifies both the animated interface agent and the facilitator. The animated interface agent is responsible for motivating students to participate in more of the discussions and to use the tools available in the virtual learning environment. These agents are described in more detail in the following subsections.

4.1 Problem Detection Agent

Problem Detection Agent (PDAg) is a reflex agent with internal state. Its internal state reflects the profiles of the students as time goes and PDAg perceives the actions performed by students. The PDAg *see* and *next* functions integration have the goal of updating the profiles of students with data about the use of the collaborative tools available in the environment. In the proposed agent based approach, for each action performed by the student in the environment, he is punctuated in the internal state with a score based on a table whose values are defined previously. Only for facilitating the comprehension of this paper, we have defined the scores presented in Table 1.

Table 1 Table of scores for participation

Interaction	Score
Message posted by chat	5
E-mail sent	20
Message sent to discussion list	10
Interest group created	30
Message sent to interest group	10

Thus, for each action defined in Table 1, in the internal environment state the student is scored with the scores defined in the table. This information will be used later by the the PDAg action function to perform the passive students detection.

The PDAg action function is responsible for detecting passive students in the process of problem based learning. The steps of the passive students detection process performed by that function are described in Algorithm 1.

How can be seen in Algorithm 1, initially it is calculated the average of students participation, eliminating the outliers. The outliers detection is performed by calculating the median values of the series. The discrepancy is eliminated in accordance with a predetermined threshold for the median. Then the average of the remaining values is calculated, which better reflect the series trend. Based on the average of the remaining values, it is possible detect a passive student out of threshold predetermined by the facilitator. Considering, for example: passive detection $threshold = 0:15$, $series = (130;140;20;135)$, $lowerbound = 0:2$ and $upperbound = 1:5$; the student with score equal to 20 would be detected as a passive student.

4.2 Animated Interface Agent

The Animated Interface Agent was implemented for communication with students. This agent aims to motivate students to participate more in discussions and use collaborative tools. It was shown in [6] that interface characters can have a positive impact on students' interactions with learning environments. This agent generates more reliance because it is socio-affective, i.e., it is able to express emotions through animations, gestures and known representations, stressing its social features.

The animated agent continuously communicates with the PDAg and requests the list of passive students that have been detected. Then the animated agent checks

Algorithm 1 Action Algorithm

```
Considering :
    student : student solving the same problem in PBL
    student_group : students set
    series : set of scores in a students group
    e : series element
    c1, c2 : central elements of the series
    order_number(series) : function to order values in ascending order
    quantity_values : number of elements of the series
    lower_bound, upper_bound : upper and lower limits for outliers
    passive_detection_threshold : detection threshold of passive students
    passive_list : stores result of the algorithm
for all student_groups do
    order_number(series)
    if mod(quantity_values/2) = 0 then
        median = (c1 + c2)/2
    else
        median_position = (quantity_values + 1)/2
        median = series(median_position)
    end if
    for all e ∈ series do
        if e/median > lower_bound ∧ e/mediana < upper_bound then
            sum = sum + e
            number_elements = number_elements + 1
        end if
    end for
    average = sum/number_elements
    for all student ∈ student_group do
        if e/average < passive_detection_threshold then
            passive_list = student
        end if
    end for
end for
```

whether students on this list are online. If a student is online, the animated interface agent will try to motivate him with phrases and gestures.

5 Final Remarks

In this paper, we described the implementation of an agent-based approach for detecting passive students in problem-based learning in virtual learning environments. The proposed solution aims to make the learning environment more pro-active, thus improving the learning process. The agent-based approach presented in this paper can be ported to any virtual learning environment because it was designed and implemented as a layer of software independent of the application. In future work, we intend to do a case study with a class of students in computer science course to see the impact of the proposed agent-based approach on the motivation levels of students. This case study will also aim to obtain feedback to improve the motivation strategy that is used by the animated agent because the problem of how to motivate a student to use a virtual learning environment to solve a problem, without additionally frustrating the student, is a complex task even for a human agent.

References

1. Azevedo, H., Scalabrin, E.E.: A human collaborative learning environment using intelligent agents. In: Lin, F.O. (ed.) Designing distributed Learning Environments with Intelligent Software Agents, pp. 1–32. Information Science Publishing, United Kingdom (2005)
2. Hmelo-Silver, C.: Problem-Based Learning: What and How Do Students Learn? Educational Psychology Review 16(3) (2004)
3. Hmelo-Silver, C., Barrows, H.: Goals and strategies of a problem-based learning facilitator. The Interdisciplinary Journal of Problem-based Learning 1(1), 21–39 (2006), http://docs.lib.purdue.edu/ijpbl/vol1/iss1/4
4. Moisil, I., Pah, I., Barbat, B., Popa, E.: Socio-cultural modelling of the student as the main actor of a virtual learning environment. In: Proc. of the 8th WSEAS Int. Conf. on Mathematical Methods and Computational Techniques in Electrical Engineering, vol. 119 (2006)
5. Otake, M., Fukano, R., Sako, S., Sugi, M., Kotani, K., Hayashi, J., Noguchi, H., Yoneda, R., Taura, K., Otsu, N., et al.: Autonomous collaborative environment for project-based learning. Robotics and Autonomous Systems (2007)
6. Reategui, E., Boff, E., Campbell, J.A.: Using virtual characters in personalized recommendations. In: SIGCSE 2007: Proceedings of the 38th SIGCSE technical symposium on Computer science education, pp. 180–184. ACM, New York (2007), http://doi.acm.org/10.1145/1227310.1227374
7. Russell, R., Norvig, P.: Artificial Intelligence, a Modern Approach, 2nd edn. Prentice Hall, Englewood Cliffs (2004)
8. Savery, J.: Overview of problem-based learning: Definitions and distinctions. The Interdisciplinary Journal of Problem-based Learning 1(1), 9–20 (2006)
9. Sendag, S., Ferhan Odabas, H.: Effects of an online problem based learning course on content knowledge acquisition and critical thinking skills. Computers & Education (2009)
10. Strobel, J., van Barneveld, A.: When is PBL More Effective? A Meta-synthesis of Meta-analyses Comparing PBL to Conventional Classrooms. Interdisciplinary Journal of Problem-based Learning 3(1), 4 (2009)
11. Tseng, K., Chiang, F., Hsu, W.: Interactive processes and learning attitudes in a web-based problem-based learning (PBL) platform. Computers in Human Behavior 24(3), 940–955 (2008)
12. Vizcaíno, A.: A simulated student can improve collaborative learning. International Journal of Artificial Intelligence in Education 15(1), 3–40 (2005)
13. Weiss, G.: Multiagent systems: a modern approach to distributed artificial intelligence. The MIT press, Cambridge (1999)

Context-Aware Agents for People Detection and Stereoscopic Analysis

Sara Rodríguez, Juan F. De Paz, Pablo Sánchez, and Juan M. Corchado

Abstract. This paper presents a multiagent system that can process stereoscopic images and detect people with a stereo camera. In the first of two phases, the system creates a model of the environment using a disparity map. It can be constructed in real time, even if there are moving objects present in the area (such as people passing by). In the second phase, the system is able to detect people by combining a series of novel techniques. A multi-agent system (MAS) is used to deal with the problem. The system is based on cooperative and distributed mechanisms and was tested under different conditions and environments.

Keywords: Multi-agent systems, stereo processing, people detection, SAD, HOG.

1 Introduction

For many years, the scientific community has demonstrated an increasing interest in the study of artificial vision. Image processing applications are varied and include such aspects as remote control, the analysis of biomedical images, character recognition, virtual reality applications, and enhanced reality in collaborative systems, among others. Although image analysis and people detection is a well explored topic, the use of multiagent technology in this area has become the focal point of important interest [2][13]. The availability of commercial hardware to solve the lowlevel problems of stereo processing has turned them into an attractive sensor to develop intelligent systems. Stereo vision provides a type of information that offers several advantages in the development of human–machine applications.

This paper presents a system that is capable of processing stereoscopic images and detecting people with a stereo camera. In the first phase, the system creates a model of the environment using a disparity map. The model can be constructed in real time, even when there are moving objects present in the area (such as people passing by). For this reason, it is an appropriate tool for using on mobile devices

Sara Rodríguez · Juan F. De Paz · Pablo Sánchez · Juan M. Corchado
Departamento Informática y Automática
Universidad de Salamanca
Plaza de la Merced s/n, 37008, Salamanca, Spain
e-mail: {srg,fcofds,pablomanzano,corchado}@usal.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 173–181.
springerlink.com

(such as mobile robots). Our system analyzes and detects people by combining a series of novel techniques and raw images detectors such as Sum of Absolute Differences (SAD) or Gradient Orientation Histograms (HOG).

The remainder of this paper is structured as follows. Section 2 explains the basis of the background modelling and foreground techniques. Section 3 shows how the multiagent systems performs stereo processing and people detection. Section 4 presents the experiment carried out, while Section 5 presents the conclusions and suggests possible future work.

2 Background

The primary concepts used in the development of this system include stereoscopic data handling, as well as all related analysis and detection processes, and multiagent technology.

Traditionally, the use of stereoscopy as a technique for reconstructing images has dealt with two problems. Using a two-dimensional pair of images with spatial coordinates (u,v), the left image (L) and right image (R), *the correspondence problem* attempts to find which two pixels $m_L(u_L,v_L)$ from the left image and $m_R(u_R,v_R)$ from the right image correspond to the same pixel M in three-dimensional space (X,Y,Z). Once these pixels have been found, *the reconstruction problem* attempts to find the coordinates for pixel M[9]. Regarding the problem for obtaining correspondence, there are several strategies that can be classified in different ways [6]. The disparity calculation allows us to obtain the depth for each of the pixels on the image, obtaining one single image as the disparity map. Given that there is a direct correlation between the depth of the objects in an image and the disparity with a stereo pair, we can use the information from the disparity map as relative values for the depth of the objects. One of the most simple techniques is the Sum of Absolute Differences (SAD), since it operates exclusively with whole numbers. The bookstore that was used in the project (Triclops SDK [12]) establishes a correspondence between the images using this technique.

Regarding the use of stereo vision to detect people, the human form as been shown to be a difficult “object” to detect because of the significant variability in its appearance, clothes, and lighting conditions. The first thing that needs to be done is to detect a set of general characteristics inherent to the human form that can be readily identifiable, even under difficult circumstances with difficult lighting conditions [11] [10]. The present study will use the Histogram of Oriented Gradients (HOG) [4]. The fundamental idea is that the appearance of objects and the shape of an image can be described by the distribution of gradient intensity or direction. The application of these descriptors can be achieved by dividing the image into small connected regions, called cells. A histogram of oriented gradients is compiled for every cell and the pixels contained within each one. The application of these histograms represents the descriptor [5] [4]. The HOG descriptor has several advantages over other descriptor methods. As Dalal and Triggs [5] observed, this descriptor maintain an almost vertical position. The HOG descriptor is then especially suited for detecting humans in images [5] [4].

Finally, the use of deliberative BDI (*Belief, Desire, Intention*) agents [3][13] is essential in the development of the platform we are proposing. Apparently, the human visual system deals with a high level of specialization when it comes to classifying and processing the visual information that it receives, such as reconstructing an image by texture, shadow, depth, etc. Computationally, it is difficult to compete with such specialization and separate from an image only the relevant information for any particular purpose. In response to this problem, we propose implementing an algorithm over a distributed agent-based architecture that will allow visual information contained in an image to be processed in real time. Because the system is capable of generating knowledge and experience, the effort involved in programming multiple tasks will also be reduced since it would only be necessary to specify overall objectives, allowing the agents to cooperate and achieve the stated objectives.

3 Our Approach

The different process are implemented over a distributed agent-based architecture, which allows it to run tasks in parallel using each service as an independent processing unit. The architecture would allow a stereoscopic image processing system to carry out its own phases, which could be distributed among the agents. This way each of the tasks, including data gathering, proprocessing, filtering and reconstruction, as well as human form detection, could be carried out. In addition to the specialized agents, there is an agent platform for monitoring and supervising the correct functioning of the system. A description and initial proposal for this architecture can be found in [13]. The system is comprised of a set of agents with defined roles that share information and services. The analysis of images supposes a complex process where each agent executes its task with the information available at each moment.

The data obtained from the stereoscopic camera are entered into the system and shared between the agents that will use specific services to process the data (filtering, preprocessing, disparity analysis, etc.) [13]. The system exits can be located in either the high-density disparity map that is obtained from the distance between the camera and the objects, the numerical representation of these distances, their three-dimensional representation in real time, and/or the detection of human forms in the specified area.

This section presents the core of stereo processing and detection processing. It is divided into two parts. The first part explains the basis of stereo calculation and how the 3D points captured by the stereo camera are translated to another reference system more appropriate for our purposes. The second part explains how the system is used to extract information about the location of the individuals.

A commercial stereo camera [13] was employed in this work because it can capture two images from slightly different positions (stereo pair) that are transferred to the computer to calculate a disparity image containing the points matched in both images. Knowing the extrinsic and intrinsic parameters of the stereo camera it is possible to reconstruct the three-dimensional position. The stereo calculation is made with the Triclops library[12], which defaults to a pairing algorithm based on the Sum of Absolute Differences (SAD). As it is the method used in the libraries provided by Point Grey [12], we chose to implement it, along with a proposal for optimizing the algorithm via the parallelization of the tasks by the algorithm [13].

SAD operates exclusively with whole numbers. Given a pixel with coordinates (x, y) in the left image, a correlation index C(x, y, s) is calculated for each displacement s for the correlation window in the right image. To calculate the correlation index, $C(x,y,s) = \sum_{u=-w,v=-w}^{u=w,v=w} | I_l(x+u,y+v) - I_r(x+u+s,y+v)) |$ where 2w + 1 is the size of the window centered on the pixel located at position (x, y) and I_l, I_r are the gray values for the pixels in the left and right images respectively. The disparity $d_l(x, y)$ between the left and right image pixels is defined as displacement s which minimizes the correlation index: $d_l(x,y) = arg\ min_s\ C(x,y,s)$.

Fig. 1(a) shows an example of a scene captured with our stereo camera (the image corresponds to the right camera). Fig. 1(b) shows the three-dimensional reconstruction of the scene captured using the points detected by the stereo camera. The ‘‘world’’ and camera reference systems have been superimposed in Fig. 1(b), which shows that the number of points acquired by a stereo camera can be very high (they are usually referred to as point cloud). For that reason, many authors perform a reduction of the amount of information by orthogonally projecting them onto a 2D plan-view map [7].

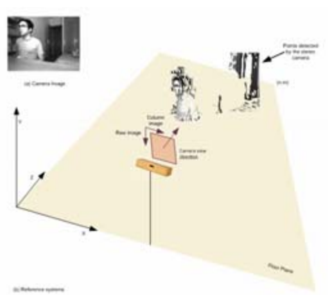


Fig. 1 (a) Image of the right camera captured by stereo system. (b) Three-dimensional reconstruction of the scene showing the reference systems employed.

The image analysis provides a point cloud in which each point represents a píxel in the image that indicates the position of the coordinates XYZ. The starting point of the coordinates used to represent the image is taken from the right-side reference point of the camera. The x-axis is horizontal, i.e., the axis that joins the camera's two reference points. The y-axis is the vertical axis that follows the camera's orientation. The z-axis measures the distance to the camera and is the axis that is perpendicular to the reference point.

People detection and stereo processing are treated as separate processes in this study. Every time a new image is captured, the system must first apply stereo processing to obtain the distances of the objects in the image. After that, the system can decide to apply the people detection to the same image. To achieve this goal, the HOG [4][5] algorithm, along with the classifiers used for training and validating the dataset, was applied, in addition to the test case that included the set of images taken with the stereoscopic camera.

Fig. 2 Detection phases

As explained in section 2, there are two fundamental issues regarding the detection of objects: the extraction of the most relevant characteristics, and the learning obtained from the classification [11]. Instead of using the raw image directly, it is common to use characteristics that are based on points, stains or gauss differences, intensities, gradients, color, textura, or a combination of these methods [8][1][14]. The developed system provides a set of services used specifically for extracting relevant characteristics [8][1][14], that comprise the base of the HOG descriptor and are used as the initial step of detection[13]. The software specifically uses the R-HOG block descriptor, which superimposes a square or rectangular block on the network cells. Each block is normalized independently. The primary phases carried out by the system during its detection process can be seen in Figure 2.

• During the first and second phase, the data are input from the camera, and the filtering process required for reducing the lighting and shading effects are applied. The third stage computes first order image gradients. These capture contour, silhouette and some texture information, while providing further resistance to illumination variations. The system provides the most appropriate form detector, according to Sobel, Canny, etc. [14][1][13].

• The fourth stage aims to produce an encoding that is sensitive to local image content while remaining resistant to small changes in pose or appearance. The adopted method combines radient orientation information locally in the same way as the SIFT[10] feature. The image window is divided into small spatial regions, called “cells”. For each cell we accumulate a local 1 ‑ D histogram of gradient or edge orientations over all the pixels in the cell. This combined cell ‑ level 1 ‑ D histogram forms the basic “orientation histogram” representation. Each orientation histogram divides the gradient angle range into a fixed number of predetermined bins. The gradient magnitudes of the pixels in the cell are used to vote into the orientation histogram.

• The fifth stage computes normalization, which takes local groups of cells and contrast normalizes their overall responses before passing to next stage. Normalization introduces better invariance to illumination, shadowing, and edge contrast. It is performed by accumulating a measure of local histogram “energy” over local groups of cells that we call “blocks”. The result is used to normalize each cell in the block. Typically each individual cell is shared between several blocks, but its normalizations are block dependent and thus different. As a result, the cell appears several times in the final output vector with different normalizations. There are two main blocks: rectangular R-HOG blocks and circular C-blocks. The R-HOG blocks are generally square and represented by three parameters: the number of cells per each block, the number of pixels per each cell, and the number of channels per cell histogram. The their experiment on human detection, Dalal and Triggs [4], determined the optimal parameters to be: 3×3 blocks of 6×6 pixel cells, with 10.4% miss rate. Four different block normalization schemes were evaluated for each of the above HOG [5]. If v ix the non-normalized vector that contains all of the histograms for one block, $\|v\|_K$ the k - norm for k=1, 2, and ε a small normalization constant to avoid division by zero. The four schemes are: (i) L2 - norm, $v \leftarrow v / \sqrt{\| v \|_2^2 + \varepsilon^2}$;(ii) L2 - Hys, L2 - norm followed by clipping (limiting the maximum values of **v** to 0.2) and renormalizing; (iii) L1 - norm, $v \leftarrow v / (\| v \|_1 + \varepsilon)$;(iv) L1 - sqrt, $v \leftarrow v / \sqrt{v / (\| v \|_1 + \varepsilon)}$. L1 - norm followed by square root essentially treats the descriptor vectors as probability distributions, using the Bhattacharya distance between them.

• The next step collects the HOG descriptors from all blocks of a dense overlapping grid of blocks covering the detection window into a combined feature vector for use in the window classifier.

• The final step in recognizing forms using HOG is to feed the descriptors with some recognition system based on supervised learning. The SVM classifier is a binary classifier that looks for the optimalhyperplane as a decision making function. Once it has been trained with the images that are contained in a particular object, the SVM classifier can make decisions with regards to the presence of an object, such as a human being, in the testing images (which are different from the training images). To this end, a modified SVMLight packet from Dalal and Triggs [5] was used, obtaining the results shown in the following figures, in which the human form can be detected in different positions.

Fig. 3 Results in the process of detection human forms

4 Experiment and Results

In explaining the model we have provided examples of its performance. A broader experimentation was done to test processing and detection of different people under different lighting conditions and different distances from the stereo vision system. We employed 640x480 sized images and sub-pixel interpolation to enhance the precision in the stereo calculation. The operation frequency of our system is near 10 Hz on a 3.2 Ghz Pentium IV computer running with Windows XP. The camera has the following characteristics [12]: 640x480 pixel sensors, monochrome, 3.8mm focal distance, capable of capturing 48 photograms per second, 120mm line base, 6 pine IEEE-1394 (FireWire) interface connection. The images were taken from a height of 1.6m with a 6fps velocity, obtaining approximately 400M coded data in AVI and PGM format (16 bits per image). More than the fifty percent of the computing time is dedicated to image capturing and stereo computation (about 50 ms) and the rest to detection (about 40 ms). This indicates that the proposed system is fast enough to be used in real time applications. In order to evaluate the system's capacity, a variety of tests were performed. The system improved the processing capability compared to other centralized systems, given that the distributed agents approach makes it possible to carry out processing tasks individually and with different techniques (selection of edge detectors, filters, final objectives: obtaining distance, detecting forms). Because the system is perfectly modulized, the tasks can be carried out simultaneously and or in a distributed manner. For computing disparity, we used a stereo algorithm that allowed for real-time computation of dense disparity maps [13]. Standard stereo calibration and external calibration were applied the first time the system was installed in the environment. Then the system can work without any manual configuration as long as the camera settings (e.g., camera position in the environment, lenses, focal lengths) are not changed. The system has been tested in many situations and different conditions. Here we report a summary of different experimental results performed during the course of our research.

Table 1 Accuracy results

Position	Distance	Angle	Avg. Err		Std. Dev.	
			Natural	Fluorescent	Natural	Fluorescent
P1	3.20 m	00º	62 mm	62 mm	74 mm	74 mm
P2	3.00 m	00º	60 mm	60 mm	58 mm	58 mm
P3	2.50 m	00º	54 mm	54 mm	28 mm	28 mm
P4	3.20 m	15 º	85 mm	85 mm	96 mm	96 mm
P5	3.00 m	22 º	81 mm	83 mm	82 mm	84 mm
P6	2.50 m	32 º	77 mm	83 mm	60 mm	64 mm
P7	3.20 m	-15 º	81 mm	81 mm	99 mm	99 mm
P8	3.00 m	-22 º	81 mm	83 mm	90 mm	92 mm
P9	2.50 m	-32 º	77 mm	83 mm	62 mm	66 mm

To measure the precision of the system we marked 9 positions in the environment at different distances, illumination and angles from the camera, and measured the distance returned by the system of a person standing at these positions. Although this error analysis is affected by imprecise positioning of the person on the markers, the results of our experiments, in Table 1 averaging 40 measurements for each position, show a precision in localization (i.e., average error) of less than 10 cm, but with a high standard deviation, which denotes the difficulty of obtaining a precise measurement. However, this precision is usually sufficient for many applications, like the one considered in domestic scenarios[13].

5 Conclusions and Future Work

This paper has presented a stereo processing system that integrates several capabilities into an effective and efficient multiagent platform: stereo image processing, distance calculation, real time graphical representation of depth, the identification of elements found within the area and human detection. Experimental results show good performance and high robustness to many problems: shadows, global illumination changes, background changes, people not moving, etc. Two major lines of investigation should be followed in the future in order to extend the applicability of the proposed system: outdoor environments and larger, more crowded areas. In the first case, it is necessary to consider more sophisticated techniques for background modelling and updates, such as multi-modal representation for the background and dynamic thresholding for background subtraction. In following the second line of investigation, we plan to include the system in a bigger architecture that could deal with large and crowded environments, which is probably beyond the scope of this method. This bigger architecture would control and autonomous mobile robot and be tested for its suitability for human–machine applications that require the ability to operate in real time and in changing conditions.

Acknowledgements. This research has been partially supported by the project PET2008_0036.

References

1. Canny, J.: A computational approach to edge detection. IEEE Trans. Pattern Analysis and Machine Intelligence 8(6), 679–698 (1986)
2. Castanedo, F., García, J., Patricio, M.A., Molina, J.M.: Designing a Visual Sensor Network Using a Multi-agent Architecture, ASC 978-3-642-00486-5, pp. 430–439 (2009)
3. Corchado, J.M., Glez-Bedia, M., de Paz, Y., Bajo, J., de Paz, J.F.: Replanning mechanism for deliberative agents in dynamic changing environments. Computational Intelligence 24(2), 77–107 (2008)
4. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. In: IEEE Conference Computer Vision and Pattern Recognition, USA, pp. 886–893 (2005)
5. Dalal, N., Triggs, B., Schmid, C.: Human detection using oriented histograms of flow and appearance. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3952, pp. 428–441. Springer, Heidelberg (2006)
6. Dhond, U.R., Aggarwal, J.K.: Structure From Stereo - A Review. IEEE Trans. on Systems, Man and Cybernetics 19(6) (November/December 1989)
7. Harville, M.: Stereo person tracking with adaptive plan-view templates of height and occupancy statistics. Image and Vision Computing 2, 127–142 (2004)
8. Lindeberg, T.: Feature detection with automatic scale selection. International Journal of Computer Vision 30, 77–116 (1998)
9. López-Valles, J.M., et al.: Revista Iberoamericana de Inteligencia Artificial 9(27), 35–62 (2005), ISSN: 1137-3601

10. Lowe, D.G.: Object recognition from local scale-invariant features. In: Int. Conf. on Computer Vision, vol. 2, pp. 1150–1157 (1999), doi: 10.1109/ICCCV.1999.790410
11. Pedersoli, M., Gonzàlez, J., Chakraborty, B., Villanueva, J.J.: Enhancing Real-Time Human Detection Based on Histograms of Oriented Gradients. In: Computer Recognition Systems 2. ASC. Springer, Berlin (2007)
12. Point Grey Research Inc. (2009), http://www.ptgrey.com/
13. Rodríguez, S., De Paz, J.F., Bajo, J., Tapia, D.I., Pérez, B.: Stereo-MAS: Multi-Agent System for Image Stereo Processing. In: Cabestany, J., Sandoval, F., Prieto, A., Corchado, J.M., et al. (eds.) IWANN 2009. LNCS (LNAI), vol. 5517, pp. 1256–1263. Springer, Heidelberg (2009)
14. Sobel, I., Feldman, G.: A 3x3 Isotropic Gradient Operator for Image Processing, presentado en la conferencia Stanford Artificial Project (1968)

Analysis and Design of a SOA-Based Multi-agent Architecture

Dante I. Tapia, Ricardo S. Alonso, Carolina Zato,
Oscar Gil, and Fernando De la Prieta

Abstract. One of the most prevalent approaches among distributed architectures is Multi-Agent Systems. The agents have characteristics such as autonomy, reasoning, reactivity, social abilities and pro-activity which make them appropriate for developing distributed systems based on highly dynamic scenarios. Nevertheless, the development of a multi-agent system can be an extensive and delicate process. During this process, it is convenient to utilize Agent-Oriented Software Engineering (AOSE) tools. Such tools facilitate and improve the engineering process, achieving models that are more detailed and closer to the multi-agent systems implementation. This paper presents the analysis and design of a *Flexible and User Services Oriented Multi-agent Architecture* (FUSION@). This is a new architecture that integrates intelligent agents with a service-oriented approach to facilitate and optimize the development of highly dynamic distributed systems.

Keywords: Distributed Architectures, Multi-Agent Systems, Service-Oriented Architectures, Agent-Oriented Software Engineering, Ontology Design.

1 Introduction

The development of dynamic and distributed software usually requires creating increasingly complex and flexible applications. In some cases, applications require similar functionalities already implemented into other systems which are not always compatible. At this point, developers can face this problem through two options: re-use functionalities already implemented into other systems; or re-deploy the capabilities required. While the first option is more adequate in the long-run, the second one is the most chosen by developers. However, the latter is a poorly scalable and flexible model with reduced response to change, in which applications are designed from the outset as independent software islands.

Dante I. Tapia · Ricardo S. Alonso · Carolina Zato · Oscar Gil · Fernando De la Prieta
Departamento de Informática y Automática, Universidad de Salamanca,
Plaza de la Merced, S/N, 37008, Salamanca, Spain
e-mail: {dantetapia,ralorin,carol_zato,fer,oscar.gil}@usal.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 183–190.
springerlink.com

For these reasons, it is necessary to develop new functional architectures capable of providing adaptable and compatible frameworks. A functional architecture defines the physical and logical structure of the components that make up a system, as well as the interactions between those components. There are Service-Oriented Architectures (SOA) [1] and agent frameworks and platforms which provide tools for developing distributed systems and Multi-Agent Systems (MAS) [2]. However, these tools do not solve by themselves which the development of current systems needs. Therefore, it is necessary to develop innovative solutions that integrate different approaches to create flexible and adaptable systems.

One of the most prevalent alternatives in distributed architectures is agent and multi-agent systems which can help to distribute resources and reduce the central unit tasks [3]. There are several agent frameworks and platforms, such as OAA [4], RETSINA [5] and JADE [6], which provide a wide range of tools for developing distributed multi-agent systems. An agent-based architecture provides more flexible ways to move functions to where actions are needed, thus obtaining better responses at execution time, autonomy, services continuity, and superior levels of flexibility and scalability than centralized architectures [2]. Additionally, the programming effort is reduced because it is only necessary to specify global objectives so that agents cooperate in solving problems and reaching specific goals, thus giving the systems the ability to generate knowledge and experience. Unfortunately, the difficulty in developing a multi-agent architecture is higher because there are no specialized programming tools to develop agents [7].

Furthermore, the development of a distributed MAS can be an extensive and delicate process. During this process, it is convenient to utilize Agent-Oriented Software Engineering (AOSE) tools. Among this tools, it can be emphasized the Gaia methodology [8] and the SysML modeling language [9]. Such tools facilitate and improve the engineering process, thus achieving models that are more detailed and closer to the MAS implementation. In addition, the use of ontologies provides a powerful tool for knowledge sharing, re-use and interoperability [10].

In the next section, the specific problem description that essentially motivated the development of FUSION@ is presented. Section 3 describes the main characteristics of this architecture and explains some of its components, making an emphasis on the analysis and design process of this development. Finally, section 4 presents the results and conclusions obtained.

2 Problem Description

Distributed architectures like SOA or MAS consider integration and performance aspects that must be taken into account when functionalities are created outside the system. Distributed architectures are aimed at the interoperability between different systems, distribution of resources, and the lack of dependency of programming languages [1]. Functionalities are linked by means of standard communication protocols that must be used by applications in order to share resources in the network [3]. The compatibility and management of messages between functionalities is an important and complex element in any of these approaches.

Among the existing agent frameworks and platforms we have OAA, RETSINA and JADE. Nevertheless, integration is not always achieved because of the incompatibility among them. The integration and interoperability of agents and multi-agent systems with SOA and Web Services approaches has been recently explored [3]. Some developments are centered on communication between these models, while others are centered on the integration of distributed services into the structure of the agents. Bonino da Silva *et al.* (2007) propose merging multi-agent techniques with semantic Web Services to enable dynamic, context-aware service composition. Ricci *et al.* (2007) have developed a Java-based framework to create SOA and Web Services compliant applications, which are modeled as agents.

When designing MAS it is necessary to utilize an AOSE process. There are several AOSE methodologies, languages and tools, such as Gaia, Tropos, MESSAGE, AUML, SysML, etc. Although many of them have important limitations and there are not specialized development tools, the use of AOSE tools vastly facilitates the development of MAS. In the development presented in this paper, it has been chosen the Gaia [8] and the System Modeling Language (SysML) [9] tools, because their integration allows obtaining models that are very closer to the systems implementation in a relatively fast and efficient way. Gaia is a simple and very general methodology focused on the analysis and design of MAS. The Gaia methodology is aimed at analyzing the system and designing its structure from a roles collection called organization. In Gaia the system is not detailed enough to carry out a direct implementation, but the aim is reducing the abstraction level to the point in which it can be applied traditional development techniques. After finalizing the Gaia analysis and design stages it is achieved a too high abstraction level. However, this can be an advantage for developers because there is not a dependency with a particular implementation solution. In the other hand, SysML is a robust language focused on the systems engineering and based on UML (Unified Modeling Language) [9]. Although this language is not considered to be only among the AOSE tools, it provides many characteristics that make it adequate for the design and representation of complex MAS, even taking the place of languages as AUML.

Besides using AOSE tools, it can be very useful for the analysis of systems and architectures to create ontologies that represent the entities involved on them. An ontology can be defined as an explicit specification of conceptualization [10]. This means that an ontology can represent the description of entities and relationships that an agent realizes [13]. There are many languages that allow encoding a certain ontology. The DARPA Agent Markup Language (DAML) was mainly developed for creating machine-readable representations of the Web, thus building what is known as the Semantic Web [14]. DAML syntax is based on the Resource Description Framework (RDF), a W3C recommendation for representing metadata on the web. Furthermore, the RDF Schema (RDFS) is a language created for defining ontological entities by means of RDF. However, as RDFS was not expressive enough for some purposes, the DAML project finally adopted the Ontology Inference Layer (OIL), which provides a fast reasoning support. Nevertheless, OIL has some restrictions that make it not suitable for some applications [10]. This way, DAML+OIL led to the Web Ontology Language (OWL), with several versions available: OWL Lite, OWL DL (Description Logic) and OWL Full.

The analysis and design process of FUSION@ using Gaia and SysML is presented in the following section. In addition, an ontology of the architecture has been built to model the entities implied on FUSION@ and to enhance the possibilities of sharing the knowledge involved on them.

3 Analysis and Design of FUSION@

This section is focused on describing the analysis and design process conducted during the development of FUSION@, an architecture aimed at developing intelligent highly dynamic environments that integrates intelligent agents and a service-oriented philosophy. Thus, the main objective of this paper is to describe this process, not the architecture itself, mainly because the research on FUSION@ is still under development and its description has already been published [15]. The architecture proposes several features capable of being executed in dynamic and distributed environments. These features can be implemented in devices with limited storage and processing capabilities.

Multi-agent architectures and frameworks such as OAA [4], RETSINA [5] and JADE [6] define agent-based structures to resolve distributed computational problems and facilitate user interactions. However, the communication capabilities are limited, not allowing easy integration with services and applications. On the other hand, SOA-based architectures usually do not provide intelligent computational and interactive mechanisms. FUSION@ combines both paradigms, trying to take advantage of their strengths and avoid their weaknesses. FUSION@ sets on top of existing agent frameworks by adding new layers to integrate a service-oriented approach and facilitate the distribution and management of resources. Therefore, the FUSION@ framework has been modeled following the SOA model, but adding the applications block which represents the interaction with users. FUSION@ defines four basic blocks: *Applications*, *Agent Platform*, *Services* and *Communication Protocol*. These blocks provide all the functionalities of the architecture [15].

One of the advantages of FUSION@ is that the users can access the system through distributed applications, which run on different types of devices and interfaces (e.g. computers, cell phones, PDAs). All requests and responses are handled by the agents in the platform. The agents analyze all requests and invoke the specified services either locally or remotely. Services process the requests and execute the specified tasks. Then, services send back a response with the result of the specific task. Moreover, the platform makes use of deliberative BDI (Belief, Desire, Intention) agents to select the optimal option to perform a task, so users do not need to find and specify the service to be invoked by the application. These features have been introduced in FUSION@ to create a secure communication between applications and services. There are different kinds of agents in the architecture, each one with specific roles, capabilities and characteristics. Thus, there are pre-defined agents which provide the basic functionalities of the architecture. These pre-defined agents come from the sequential stages followed using the Gaia methodology and the SysML language.

On the Gaia analysis stage it is defined the Roles Model and the Interaction Model from the characteristics and requirements of the system. This way, it is defined the roles that collaborate in the system, as well as the protocols and activities that must carry out each of the roles. The Gaia high-level stage includes the Agent Model, the Services Model and the Acquaintance Model. The Agent Model allows determining what agents should implement the roles defined before. The Acquaintance Model lets the designer realize the relationships existing among the agents to be implemented. Fig. 1 shows two Gaia analysis and design models of FUSION@. Fig. 1a depicts the Role Model for the Role Administrator, while Fig. 1b shows the Interaction Model of the processing of a service request on which three roles are involved: Application Communicator, Administrator and Service Communicator. From the Acquaintance Model we obtain the pre-defined agents: *Admin*, *Directory*, *CommApp*, *CommServ*, *Interface*, *Security* and *Supervisor* [15].

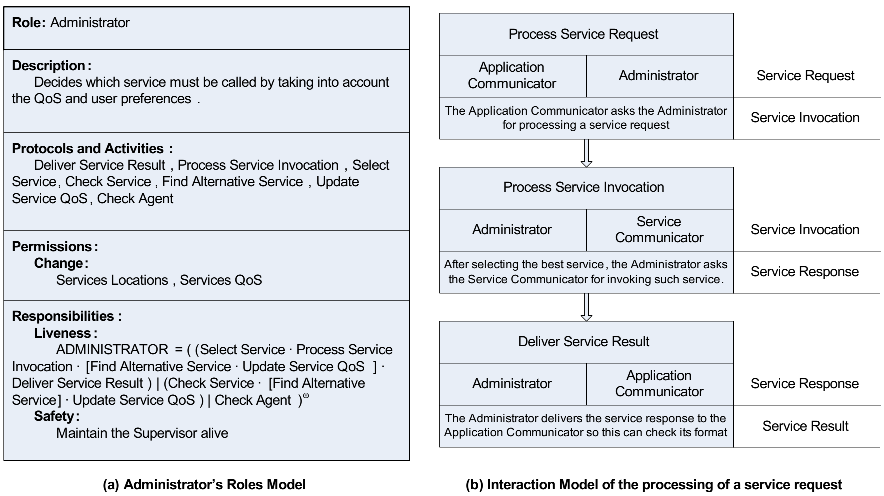


Fig. 1 Administrator's Roles Model and the Interaction Model for a service request processing

After the analysis and high-level design phases of the Gaia methodology, designers may go on analyzing the results and carry out a low-level design by means of SysML. This stage provides the Blocks Definition Diagrams, the Sequence Diagrams, the Interaction Diagrams and the State Diagrams. Each Blocks Definition Diagram represents an agent on the system, describing its capacities and services that must implement, as can be seen in Fig. 2 for the Admin Agent. Sequence Diagrams and Interaction Diagrams represent interactions among agents, the first over the time and the latter emphasizing the associations existing among them. Fig. 3 shows the State Diagram of the Admin Agent. Thus, Fig. 3 describes the possible states on which the Admin Agent can be and what events make the agent to jump from one state to another. As can be seen, this information is very useful for the developers who have to finally encode the agents.

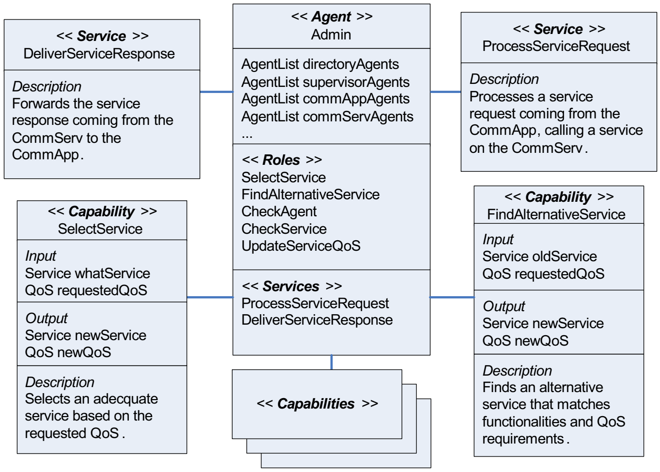


Fig. 2 Admin Agent's SysML Block Definition Diagram

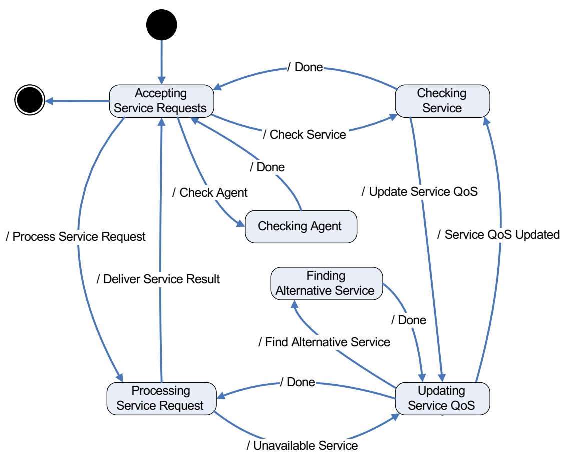


Fig. 3 Admin Agent's State Diagram

After analyzing and designing the architecture, it is possible to describe its components by means of a certain ontological language. To describe the components of FUSION@, the OWL language has been chosen, making use of the RDF/XML rendering. This choice has been motivated by the fact that OWL is especially designed for being used on the Web and can utilize the RDF and XML standards. This will be very useful for an architecture that joins Web services and intelligent agents, so it is intended to manage knowledge about itself in the future. This is a fragment of the resulting ontology[1]:

[1] For the whole ontology of FUSION@, please contact with BISITE at http://bisite.usal.es

```
<owl:Class rdf:about="#adminAgent">
   <owl:equivalentClass>
      <owl:Class>
         <owl:intersectionOf
rdf:parseType="Collection">
            <rdf:Description rdf:about="#agent"/>
            <rdf:Description
rdf:about="#administrator"/>
         </owl:intersectionOf>
      </owl:Class>
   </owl:equivalentClass>
</owl:Class>
```

FUSION@ is an open architecture that allows developers to modify the structure of the agents described before. Developers can add new agent types or extend the existing ones to conform to their projects needs. However, most of the agents' functionalities should be modeled as services, releasing them from tasks that could be performed by services. All information related to services is stored into a dynamic directory which the platform uses to invoke them. Services are requested by users through applications, but all requests are managed by the platform. This provides more control and security when requesting a service because the agents can control and validate all messages sent to the services and applications. Developers are free to use any programming language, although they must follow the communication protocol based on transactions of XML (SOAP) messages.

4 Results and Conclusions

As a conclusion we can say that although FUSION@ is still under development, preliminary results demonstrate that it is adequate for building complex systems and exploiting composite services [15]. FUSION@ has laid the groundwork to boost and optimize the development of future projects and systems that combine the flexibility of a SOA approach with the intelligence provided by agents. The distributed approach of FUSION@ optimizes usability and performance because it can be obtained lighter agents by modeling the systems' functionalities as independent services and applications outside of the agents' structure, thus these may be used in other developments.

Thanks to the employment of AOSE tools such Gaia and SysML, the analysis and design stages of the development of FUSION@ have made it possible to achieve a model close to its implementation. This have implied a shorter development time, allowing designers to focus on obtaining a well-designed distributed and dynamic architecture, rather than on encoding and debugging tasks. Furthermore, the ontology of FUSION@ will allow managing knowledge about the entities involved on it and make it possible to exchange information about agents themselves through local and remote services linked to the architecture. Our experience in the design and development of multi-agent systems has demonstrated us that the use of AOSE tools not only reduces the development time, but also improves the quality of the final system, as well as its scalability and reliability.

Acknowledgments. This work has been supported by the Ministerio de Ciencia e Innovación project PET2008_0036.

References

[1] Cerami, E.: Web Services Essentials: Distributed Applications with XML-RPC, SOAP, UDDI & WSDL, 1st edn. O'Reilly Media, Inc., Sebastopol (2002)
[2] Wooldridge, M.: An Introduction to MultiAgent Systems, 2nd edn. Wiley, Chichester (2009)
[3] Ardissono, L., Petrone, G., Segnan, M.: A conversational approach to the interaction with Web Services. Computational intelligence 20(4), 693–709 (2004)
[4] Martin, D.L., Cheyer, A.J., Moran, D.B., et al.: The open agent architecture: A framework for building distributed software systems. Applied Artificial Intelligence 13(1), 91–128 (1999)
[5] Sycara, K., Paolucci, M., Van Velsen, M., Giampapa, J.: The RETSINA MAS Infrastructure. Autonomous Agents and Multi-Agent Systems 7(1), 29–48 (2003)
[6] Bellifemine, F., Poggi, A., Rimassa, G.: JADE–A FIPA-compliant agent framework. In: Proceedings of PAAM, vol. 99, pp. 97–108 (1999)
[7] Rigole, P., Holvoet, T., Berbers, Y.: Using Jini to integrate home automation in a distributed software-system. In: Plaice, J., Kropf, P.G., Schulthess, P., Slonim, J. (eds.) DCW 2002. LNCS, vol. 2468, pp. 291–303. Springer, Heidelberg (2002)
[8] Wooldridge, M., Jennings, N.R., Kinny, D.: The Gaia methodology for agent-oriented analysis and design. Autonomous Agents and Multi-Agent Systems 3(3), 285–312 (2000)
[9] SysML - Open Source Specification Project (Recovered, October 2009), http://www.sysml.org/
[10] Pan, J.Z., Horrocks, I.: RDFS(FA): Connecting RDF(S) and OWL DL. IEEE Transactions on Knowledge and Data Engineering 19(2), 192–206 (2007)
[11] Bonino da Silva, L.O., Ramparany, F., Dockhorn, P., Vink, P., Etter, R., Broens, T.: A service architecture for context awareness and reaction provisioning. In: IEEE Congress on Services (Services 2007), pp. 25–32 (2007)
[12] Ricci, A., Buda, C., Zaghini, N.: An agent-oriented programming model for SOA & web services. In: 5th IEEE International Conference on Industrial Informatics, pp. 1059–1064 (2007)
[13] Gruber, T.R.: A translation approach to portable ontology specifications. Knowl. Acquis. 5(2), 199–220 (1993)
[14] Hendler, J.: The Dark Side of the Semantic Web. IEEE Intelligent Systems 22(1), 2–4 (2007)
[15] Tapia, D., Bajo, J., Corchado, J.: Distributing Functionalities in a SOA-Based Multiagent Architecture. In: 7th International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS 2009), pp. 20–29 (2009)

A Multiagent Scheduling Repair Method for Disruption Management in Complex Socio-technical Organizations

Sebastien Fournier and Alain Ferrarini

Abstract. In order to minimize disturbance impact in complex organizations, a distributed scheduling repair method based on agent modelling approach and new cooperative repair behaviours between modelled actors are proposed. Repair operations and repair strategies are redefined, and illustrated on a case study in organization of building site, where the proposed multiagent model is isomorphic to the actual organization. The resulting multiagent system operates the repair behaviours in order to determine the solution which minimizes the disturbance impact on the system using JADE platform.

1 Introduction

Within a more competitive financial and economic environment, the survival of companies request better effort and harshness in management of operations planning and organization. Moreover, the current economic context leads companies to deal with highly restrictive delays and labour fall (Clausen et al. 2001). In this context, as disturbances lead to important cost increase, disturbances management becomes a major issue to tackle in management of socio-technical systems. Such socio-technical organization problems are mostly complex according to their topographic distribution and the variety of tasks to be realized. This complexity increases difficulty of solving organizational problems especially on tasks planning, where the stability of the published planning is a key factor for human workers. It also highlights the interest of helping decision makers in disruption management of pre-established planning.

1.1 Context, Problematic and Objectives of the Work

Decision making in planning, site organization and management, becomes then closer to the production operations themselves. This trend can be noticed in large

Sebastien Fournier · Alain Ferrarini
LSIS UMR CNRS 6168 – Paul Cézanne University,
Domaine Universitaire de St Jérôme, Marseille 13 397, France
Tel.: +33 (0)4 91 05 60 61; Fax: +33 (0)4 91 05 60 33
e-mail: sebastien.fournier@lsis.org, alain.ferrarini@lsis.org

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 191–199.
springerlink.com

companies as well as in small business, for workshops, supply chains and also for buildings site organizations.

In order to minimize disturbances impact, a new approach is proposed in this paper based on agent model and agent-based simulation. This approach implements an agent-based modelling of socio-technical organizations, and run a distributed cooperative problem solving method. The operationalization of the multiagent system leads to the proposal of solutions minimizing disruption impacts, through multiagent simulation of recovery decisional processes.

This method has been applied on production shops inside an enterprise and in the context of inter-enterprise level (for the management of supply chains) (Tranvouez and Ferrarini 2006), (Cauvin et al. 2009). It is based on an organizational model of industrial systems whereas all the actors and resources of the system have a sufficient decisional autonomy to manage the occurrence of disruptions as soon as possible. Actors, resources, and disruptions were defined at each level according to the problem specific needs.

Firstly, this paper proposes to extend the method to more complex rescheduling or replaning problems, in particular focusing on taking in to account inventory levels (for consumed or produced goods), multiple shared resources and operator skills management in the context of multiple production. In the aim to fully take into account such complexity, the basic axioms of the original method are redefined together with the addition of new dimensions in both tasks and disruption modelling, The extension of the approach also needs adjustments such as redefinition of repair operations and disruption management strategies. It also implies to redefine who the implied actors in the process are (and thereby who (or what) the agents represent). A second aspect is the problem modelling based on the agent paradigm. The operationalization of the agent model leads to a multiagent system that executes repair behaviours in order to determine the solution minimizing the disturbance impact. Finally, an illustration on a case study in organisation of building site is explained, and implemented using JADE.

2 Cooperative Distributed Scheduling Repair Method for Managing Disruptions in Complex Organizations

The principle of the proposed method comes from the cooperative methods of distributed solving of disruptions implemented in the areas of production and logistics. This principle extended to the specificities of this issue, is based on the development of solutions by the actors themselves for limiting the impact of disruptions throughout the site while ensuring the achievement of objectives. Rather than implementing a planning calculation, they try to repair the existing situation in the aim to limit the impact of recovering solutions on human workers.

A wide variety of different approaches to repair schedule or rescheduling have been proposed as a part of Dynamic Scheduling approach (Ouelhadj and Petrovic 2009). Schedule repairing refers to "local" adjustment of the current schedule and may be preferable for many reasons especially in the dynamic context of industrial systems (Cauvin et al. 2009). Schedule repair can be done in a centralized way

(Rabideau et al. 1999) , but agent approaches have been highlighted as being well suited to complex scheduling problems (Myashita 1998) (Cowling et al. 2000). Inherent capabilities of MAS induce reduced complexity, self-configuration, increased flexibility, and reduced costs. As a response for managing disruptions in planning or schedules, our works have thus defined strategies for responding to disruptions consisting in sequences of repair operations applied on disrupted planning.

2.1 Repair Operations

A repair operation can be defined as a local and limited modification of a previously calculated scheduling (Zweben et al. 1993). Initially these operations consisted in time shifts (right or left) or tasks permutations within one or more production plans. Repairing schedules using such repair operations is particularly suited to solve incomplete scheduling problems (not taking into account tasks) or partially deficient problems (violation of time constraints), resulting for example from disruptions of the system. Furthermore, this approach has already been applied to distributed scheduling (Neiman and Lesser 1996). These repair operations can limit the rescheduling to a minimum of actors using cooperation or negotiation between them.

As regards to disturbance management in industrial systems (from production shops to basic supply chains), we have previously defined six types of repair operations (Cauvin et al., 2009) mainly based on temporal dimension (modification of the planning) and on end product inventory level. The achievement of these repair operations is mainly based on the existence of temporal margins (i. e. Idle time of actors between two successive tasks).

Disruption management in more complex organizations, such as Building and Construction Industry, needs to define repair operations that take into account additional dimensions of this issue. The enlargement of the method and then the extension of repair operations concern more particularly the qualitative characteristic (interruptible or uninterruptible tasks) and physical attributes (resources, tools, equipment, materials, indirect materials) according to the level of their stock. Repair operations depend then on: (i) time: shifts, permutations, exchanges; (ii) level of stock of resources: local operations (picking in a local stock), then in the stock from another site (co-operation), exchanges of stock (temporal characteristic); (iii) level of performed quantities (achieving the task objective by first picking in local stocks then in external ones (co-operation)).

A "*trade*" repair operation is proposed to represent the possibility of replacing a task or at least reducing its length, with its equivalent of capacity (ie stock seen here as a capacity). Moreover, a new operation "*decompose*" has been added which generalizes the possibility of interrupting a task and thus consider it as 2 different tasks: one completed and the other to be rescheduled. Resources have been detailed (human and operative) and this concept is now applicable on more complex problems. These changes do not "disrupt" the method but rather gives new means to reduce the impact of a disruption on a system where *Resources* can *locally* or *cooperatively* change *tasks* allocated realization time in order to give some *agility* and reactivity toward the environment changes. The repair operations are implemented according to strategies adapted for processing these new dimensions.

2.2 Strategies for Solving Disruptions

Solving strategies implement sequences of repair operations in order to limit the impact of disruptions on the system actors. Each implemented strategy will evaluate the solution in order to decide (or not) of using a strategy more costly in terms of impact on the system. The evaluation of each solution is based on a multi-criterion function measuring its effectiveness (degree of success in the absorption of the disruption), its complexity (the extent of changes to the original scheduling) and its agility (decrease of time rooms for manoeuvre of actors).

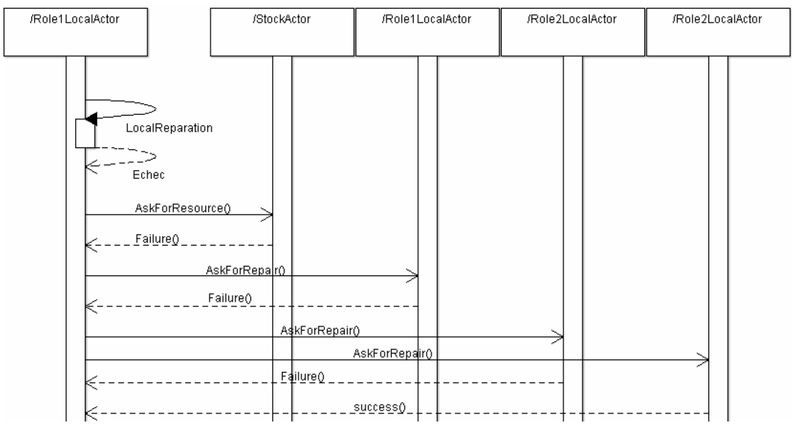


Fig. 1 Distributed solving strategies

As shown in figure 1, when a disruption occurs in a task (done by Ma1), the actor first tries to solve it by a local strategy implementing one or more operation (s) of time repair and in case of failure (after evaluation of solutions) one or more local repair operations on resources of the site. In case of failure a strategy of bilateral cooperation is first initiated and then a cooperative multilateral strategy intra site is implemented in case of failure of previous strategies. In case of failure of all local strategies, the impact of the disruption is then propagated to the sub sites simultaneously (not presented on this figure). This requires cooperation and uses the same previous strategies but within the concerned sub-site(s).

3 Agent Based Modelling for Distributed Problem Solving

The modelling paradigm previously used for simplified industrial systems considers systems as composed of autonomous entities. These entities are pro-active and defined by their behaviours (i.e. their interactions according to their roles and competencies). This paradigm can be directly extended on more complex socio-technical organization problems (for instance construction site management).

3.1 Agent Modelling

The actors involved in industries (head managers, technicians, skilled tradesman, workers, etc.) have means, resources, skills and competencies that software agents exhibit as well. All these abilities make it possible to perform tasks or functions in an individual or collective way inside an organisation. This analogy between agents and actors already used for industrial systems modelling (Nwana 1996) (Parunak 1999) (Brukner *et al* 2003) naturally led us to consider the agent-based approach as relevant for this issue. Consequently, similarities between the properties of both agents and actors enable us to make a bijection between industry actors and agents of multi-agent systems. This bijection thus generates an isomorphism of structure between industrial field and agent-based modelling (Fig. 2).

Moreover, agent-based modelling allows us to easily take into account the: (i) the natural distribution of multi-site organizations (several actors from the same trades working together, several sites managed simultaneously, etc.), (ii) the dynamic changes in the environment through the addition or suppression of resources or actors, (iii) the decisional complexity and variability of multi-sites management.

The natural ability of multi-agent systems to solve distributed problems make it possible to manage these requirements.

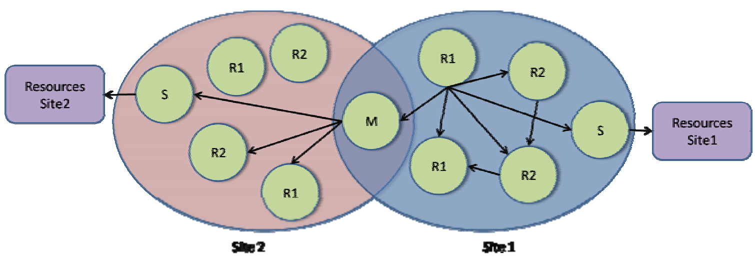


Fig. 2 Multiagent architecture

In this example (Fig.2), agents represent the actors of the two operating sites (Ri for operators, M for global manager, S for inventory managers). The global manager rules the two sites so the agent M is the gateway between the two sets of agents for the execution of inter-site cooperation strategies.

For instance, the Building and Construction site management field specifically introduces a supplementary complexity level which requires an extension of the previously defined methods. This supplementary complexity level takes its roots remarkably in a greater anthropisation of the modelled system. This fact is notably noticed in task realisation and in resources utilisation (as much in the notion of quantity as in the notion of availability) and induces the reinforcement of the temporal aspect. This anthropisation result of the main characteristic of this application field: tasks are quite exclusively realised by human (tradesmen) and not quite exclusively by machines.

Agents will represent the actors of the real system with specific abilities and knowledge to implement repair strategies. Then multiagent system makes it possible to describe more precisely how the methods could be undertaken in domain dependant models focused on the notion of actor, and the operationalisation of the multiagent system allows to simulate the decisional processes of the involved actors, and to evaluate the pertinence of the repairing solution.

3.2 Agent Behaviour Design

Each actor/agent has the same goal: achieve scheduled tasks. So they implement the same behaviour according to their skills. This behaviour, as shown in figure 3, is mainly oriented to take care of disturbances in order to achieve scheduled tasks. (1-2) When the message queue is empty, the agent try to achieve its task. If disruption occurs during the task (3) two strategies (4) are possible according to stock level: Temporal local repair or requesting for stock. If the stock level is sufficient the task can resume (5). If a local repair (6) can be done the agent modifies its planning and achieves the scheduled task. In case of previous strategies failure (7), the agent calls for help by sending messages and waiting answers (8). (9) The agent analyses the different repair solutions that it has received, accepts one (10), modifies its planning (11) and then executes the new scheduled task (12). The agent (13) may receive request for helping repair from other agents and send repair solution. If the proposed solution is accepted (14) the agent modifies its own planning.

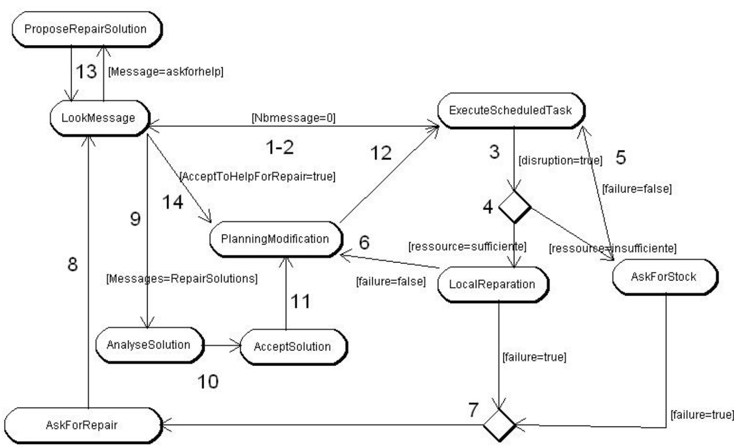


Fig. 3 Agent Behaviour

4 Illustration of the Method on the Case Study

This work is based on a specific case study: Site renovation of a small building. The implied trades are thus more of light work type than main walls works type. Figure 4 shows the different tasks that are implied in this work. A disturbance occurs during renovation of the kitchen. More precisely, the disturbance occurs during the task "installation of partition". In fact, one of the plasterboard cannot be used to create the partition.

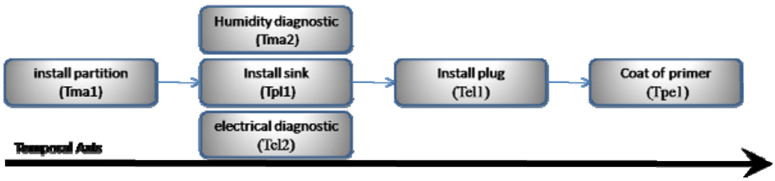


Fig. 4 Dependency graph of tasks

In order to continue the tasks needed to finish the work, it is necessary to resorb this disturbance. As evoked in previous section this approach tries to solve such problem at the more local state. In the example presented in this paper, one way to deal with the disturbance is to replace the plasterboard defective. Opportunities to solve the disturbance are as following: (i) shifts right tasks to get enough time in order to insert the delay; (ii) takes plasterboard in the second building site. The problem is then deported to the second building site; (iii) begin immediately the work on the second building site in order to gain the delay to obtain the plasterboard for the first building site.

Following principles presented in this paper, the solution must be the most local in order to avoid disturbance propagation to other system actors. In this case study, if the plasterboard is not in the stock, the more local solution is to shift tasks. Only diagnostic tasks can be executed immediately. In fact other tasks have a dependence link with "installation of partition". Furthermore, these tasks cannot be modified individually. Then, the most local solution is to realize task Tma2 before Tma1 in order to reconstruct the stock of plasterboard. The shift right operation modifies the planning. All tasks bound to "installation of partition" must be shifted to time interval dedicated to "humidity diagnostic".

5 Test Platform

The system is developed using ECLIPSE and the framework JADE. Thanks to JADE (JADE, 2007). The prototype is FIPA compliant and uses the normative ACL-FIPA communication language. Each agent execute a behaviour that defines their actions (execute a task, communicate) during the simulation.

The system deals with different kind of agents. Few of them are used for management but mainly agents represent actors of the real system. Actor agents are built on the same pattern. During the execution such agents can play different JADE's behaviours depending on agent's interaction and environment changes. The prototype gives interesting first results using local repair operations; cooperative multi-lateral strategies are currently under development.

6 Conclusion and Future Work

In this paper a method of cooperative distributed repair for planning or scheduling using multiagent system is presented. It is an extension of the approach developed for production systems and then extended to the issue of supply chain management. This extension mainly focuses on the consideration of new dimensions of problem complexity and on the development of repair planning in order to reduce the impact of the disruption. These new concepts are integrated in the management of distributed socio-technical organization and the model is illustrated on a building site organization. Future work will focus on (i) the extension of the metric for evaluating the impact of disruptions (the shift of a task that does not use a stock does not have the same cost as a task requiring stocks), (ii) the implementation on a real case using real data and the collaboration with social science researchers to better define the basis axiomatic (tasks / disruptions / measurement of the impact on workers).

References

Bruekner, S., Baumgaertel, H., Parunak, H.V.D., Vanderbok, R., Wilke, J.: Agent Models of Supply Network Dynamics: Analysis, Design and Operation. In: Harrison, T.P., Lee, H.L., Neal, J.J. (eds.) The Practice of Supply Chain Management: Where Theory and Application converge (2003)

Cauvin, A., Ferrarini, A., Tranvouez, E.: Disruption management in distributed enterprises: A multiagent modelling and simulation of cooperative recovery behaviours. International Journal of Production Economics 122, 429–439 (2009)

Clausen, J., Hansen, J., Larsen, J., Larsen, A.: Disruption management. OR/MS Today 28, 40–43 (2001)

Cowling, P., Ouelhadj, D., Petrovic, S.: Multi-agent systems for dynamic scheduling. In: Garagnani, M. (ed.) Proceedings of the nineteenth workshop of planning and scheduling of the UK, pp. 45–54. The Open University, UK (2000)

Miyashita, K.: CAMPS: a Constraint-Based Architecture for Multiagent Scheduling. Journal of Intelligent Manufacturing 9, 145–154 (1998)

Neiman, D., Lesser, R.: A Cooperative Repair Method for a Distributed Scheduling System. In: Drabble, B. (ed.) AIPS 1996: Proceedings of the Third Int. Conf. on Artificial Intelligence Planning Systems, Edinburgh, Scotland, pp. 166–173 (1996)

Nwana, N.H.: Software Agents: An Overview. Knowledge Engineering Review 11(3), 1–40 (1996)

Ouelhadj, O., Petrovic, S.: A survey of dynamic scheduling in manufacturing systems. Journal of Scheduling 12(4), 417–431 (2009)

Parunak, H.V.D.: Industrial and Practical Applications of DAI. In: Weiss, G. (ed.) Multiagents Systems: A Modern Approach to D.A.I. MIT Press, Cambridge (1999)

Rabideau, G., Knight, R., Chien, S., Fukanaga, A., Govindjee, A.: Iterative planning for spacecraft operations using the aspen system. In: Proc. 5th Int. Sym. on AI, Robotics and Automation for Space (1999)
Tranvouez, E., Ferrarini, A.: MultiAgent Modelling of Cooperative Disruption Management in Supply Chains. In: IEEE – Int. Conf. on Service System and Service Management (ICSSSM 2006), Troyes, France (2006)
Wooldridge, M., Ciancarini, P.: Agent-Oriented Software Engineering: The State of the Art. In: Ciancarini, P., Wooldridge, M.J. (eds.) AOSE 2000. LNCS, vol. 1957, pp. 1–28. Springer, Heidelberg (2001)
Yu, G., Qi, X.: Disruption management: framework models and applications. World Scientific, Singapore (2004)
Zweben, M., Davis, E., Daun, B., Deale, M.: Scheduling and Rescheduling with Iterative Repair. IEEE Trans. on Man and Cybernetics 23(6), 1588–1596 (1993)

Price Updating in Combinatorial Auctions for Coordination of Manufacturing Multiagent Systems

Juan José Lavios Villahoz, Ricardo del Olmo Martínez, Alberto Arauzo Arauzo, and José Manuel Galán Ordax

Abstract. In this work we use the paradigm of multiagent systems to solve the Job Shop problem. Job Shop problem is a particular problem of scheduling in which we try to find an schedule that optimize a objective and is subject to certain constraints. We propose a combinatorial auction mechanism to coordinate agents. The “items” to be sold are the time slots that we divide the time horizon into. In tasks scheduling problems tasks need a combination of time slots of multiple resources to do the operations. The use of auctions in which different valuations of interdependent items are considered (e.g. combinatorial auctions) is necessary. A certain time slot will be more valuated to the extent that it enables task to finish the job on time, together with the other time slot bought. Job-agents are price-takers in the model. The auctioneer fixes prices comparing the demand over a time slot of a resource with the capacity of the resource in this time slot. There are many ways to update prices (e.g. constant increase or decrease of prices, proportional to the demand, proportional to the excess of demand). Our objective is to compare the different methods of updating prices based on those of the lagrangian relaxation solving method.

Keywords: Multi-agent systems, Manufacturing Multiagent System, Auctions, Lagrangian Relaxation.

1 Multiagent Systems in Manufacturing

Scheduling problem is a decision-making problem concerning the allocation of resources to tasks over time periods and its goal is the optimization of one or several

---

Juan José Lavios Villahoz · Ricardo del Olmo Martínez · José Manuel Galán Ordax
INSISOC. Escuela Politécnica Superior,
Universidad de Burgos
Avda. Cantabria s/n, 09006, Burgos, Spain
e-mail: {jjlavios,rdelolmo,jmgalan}@ubu.es

Alberto Arauzo Arauzo
INSISOC. ETSII
Universidad de Valladolid
Pº del Cauce s/n, 47011 Valladolid, Spain
e-mail: arauzo@insisoc.org

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 201–207.
springerlink.com

objectives. Manufacturing and production systems are one of the most known fields of application of this problem, where tasks are operations of some production process and resources are machines in a workshop. (Pinedo 2008). The main features of Manufacturing scheduling problem are its highly combinatorial and dynamic nature and its practical interest for industrial applications (Shen 2002).

Multiagent Systems have proved to be an appropriate paradigm to model complex systems and they constitute a useful frame to define distributed decision-making processes. Agents are used to encapsulate physical and logical entities or even functionalities of the production system. These systems are based on the autonomy of each agent and on the interaction and negotiation among the agents. The application of this paradigm to the task scheduling problem has had a great development. A revision of the main work in this area can be found in (Shen et al. 2006), (Lee & Kim 2008) and (Ouelhadj & Petrovic 2009).

2 Market Based Coordination Mechanisms and Combinatorial Auctions

Distributed decision making is considered an alternative to pure centralized scheduling systems as it facilitates the incorporation of local objectives, preferences and constraints of each resource in the decision making process. (Kutanoglu & Wu 1999). Combination of individual problem-solving and coordination/negotiation schemes is one of the research challenges in this area. (Shen et al. 2006).

Market based allocation mechanisms are one of the most active branches of distributed task scheduling. It involves the creation of a production schedule based on the prices emerging from the bids sent by tasks. Each task proposes a bid trying to maximize its own objective. Each task has only access to its own information (objectives, preferences and constraints). The underlying idea is to allocate resources among the task by the creation of an ad-hoc market, setting prices through the search of equilibrium in an iterative process. There is no communication among agents representing tasks. Prices enable coordination of agents. Complex calculations are distributed among the participating agents, so that the problem is divided into several easier problems which can be solved in parallel. The communication overhead is due as it is limited to the exchange of bids and prices between agents and the market mechanism (Wellman et al. 2001).

We propose a combinatorial auction mechanism to coordinate agents. In tasks scheduling problems tasks need a combination of time slots of multiple resources to do the operations. The use of combinatorial auctions as a negotiation and coordination mechanism in Multiagent Systems is appropriate in problems in which different valuations of interdependent items have to be considered. In combinatorial auction participants bid for a combination of different products. The valuation of an item depends on the combination of products it belongs to (i.e. a bidder will make his valuation of an item based on which will be the other items that he will buy in the same auction.

There are different kinds of combinatorial auctions. In the single round auctions participants send their valuations over the combinations of products just one time and the auctioneer allocate the items in such a way that the global objective is maximized. The alternative to single-round auctions are the iterative combinatorial auctions, where prices are fixed after multiple rounds.

There are some advantages of using iterative combinatorial auctions rather than single-one ones. First, participants do not have to make bids over the set of all possible combinations of bids. Second, participants reveal in each iteration their private information and preferences. Third, iterative auctions are well-suited for dynamic environments (i.e. manufacturing environment) where participants and items get in and out in different moments.

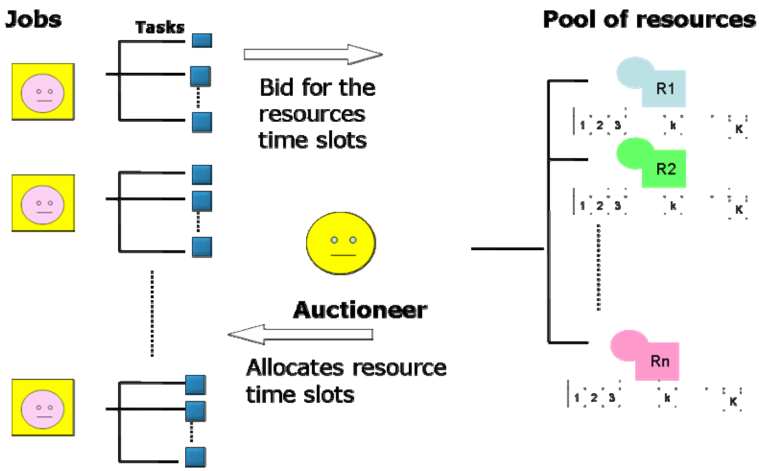


Fig. 1 Auction-based scheduling mechanism

There are two kinds of iterative combinatorial auctions: quantity-setting and price-setting. In the fist one, bidders send their valuation over the items they want to buy. The auctioneer makes a provisional allocation that depends on the submitted prices. Bidders adjust the prices in each iteration. In the second one, quantity-setting, auctioneer set the prices for each of the items of the auction. Bidders submit the bundle of items they want to get at the given price. Auctioneer adapts the prices so to balance the supply and demand. (de Vries & Vohra 2003).

3 Proposed Mechanism

Tasks scheduling problem can be modeled as an auction where time horizon is divided into slots that are sold in the auction. Tasks participate in the auctions as a bidders, trying to get the time slots of the resources that they need to perform the operation (Dewan & Joshi 2002). The mechanism will follow the main principles of distributed systems so the relevant information of the bidders, due date of the jobs, penalty for the delays will be hidden to the rest of agents (Duffie 1990). None of the agents knows which other agents do not belong to the system nor what are the goals of those agents. Prices show the preferences of other agents, and let the agent act consequently.

The mechanism proposed can be described as follows: There exists a central pool of resources (Resource-agent). Each resource has different abilities. The planning horizon of the resource is divided into time slots. These time slots are sold in an auction. There are a several jobs to be done (Job-agents). The jobs need to get the necessary resources to be finished before their due date. Job-agents bid for the time slot of the resources. They try to minimize their cost function. An agent acts as a central node (Auctioneer-agent). The bids sent by Job-agents are received by the Auctioneer-agent, who will update the prices of the slots. Once known the new prices, the Job-agents will remake their bids. This iterative process continues until prices are stabilized or a stop condition is fulfilled. Full process is detailed below.

Step 0. Initialization. Auctioneer fixes the initial prices of the time slot of the resources λ_0. Usually, it starts with a price of zero for all the time slots $\lambda_0 = 0$.

Step 1. Individual optimization. Agent-jobs have to solve an optimization problem in every iteration given the price of the time slots (λ_i). The cost function of a job is given by the sum of the penalization for the delay of the job and the price of the resource time slots used by the job. This will require evaluating all the possible alternatives and choosing among them that of minimum cost. The bid consist of the bunch of slots chosen by the Job-agent that is sent to the auctioneer.

Step 2. Evaluation of Bids. Once the Job-agents have sent their bids, the auctioneer combines all the schedules sent by the tasks and evaluate the need of capacity of every resource in every time slot.

Step 3.Construction of a feasible Schedule. The solution proposed by the Job agents is not feasible in most of cases as the demand of certain resource time slots will exceed the capacity of the resource. The auctioneer proposes a feasible schedule based on these proposals.

Step 4. Stopping criteria. The stopping criteria considered are 1) the dual gap is small (the dual gap is the difference between the best known value of the original function and the best known value of the dual function), 2) the step size to update prices is small enough 3) the maximum number of iterations is reached.

Step 5. Price updating. Price updating considers the capacity of resources and their demand in every time slot. The objective of update the prices of the time slot is to reduce the conflicts between agents. High price of a slot of time means it is in a great demand and it will penalize its use. Prices will rise when the demand of a slot is greater than its capacity, whereas they will fall if the demand of a slot is lower than its capacity. Prices can not be negative.

Job-agents are price-takers in the model. Price is fixed by an iterative process. The prices of the slots are raised or lowered by a walrasian mechanism. The auctioneer compares the demand over a time slot of a resource with the capacity of the resource in this time slot. If there is an excess demand raises prices the auctioneer raises prices and he lower them if there is an excess of capacity.

$$\lambda^{n+1} = \lambda^{n} + \Delta\lambda$$

There are many ways to update prices (e.g. constant increase or decrease of prices, proportional to the demand, proportional to the excess of demand).

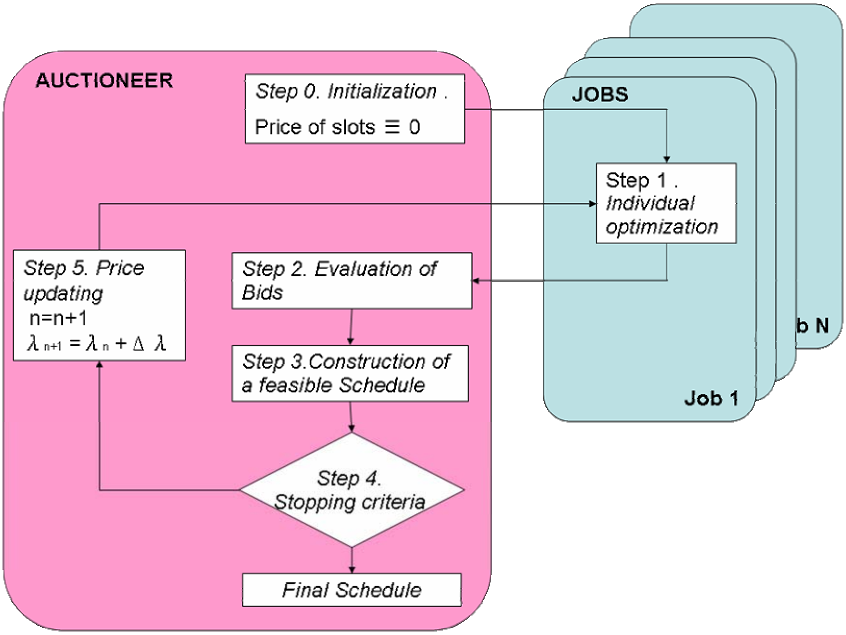


Fig. 2 Price-setting iterative combinatorial auction

4 Combinatorial Auctions and Lagrange Relaxation Techniques

There is a similarity between iterative combinatorial auction and Lagrangian Relaxation Algorithm. The update of prices process is similar to the update of Lagrange multipliers iterative process. The updating methods used in the Lagrangian Relaxation Method can be used to update the prices in the iterative auction. (Dewan & Joshi 2002). This mechanism have been used to solve different kind of task scheduling problem, e.g. job shop problem (Kaskavelis & Caramanis 1998), (Sun et al. 2006), flow shop problem (Tang et al. 2006).

Lagrangian relaxation technique relaxes the complex constraints of an optimization problem (i.e. those constraints which make the problem hard to solve), making the problem to optimize easier. The group of relaxed constraints is incorporated to the objective function in such a way that the restrictions that are not met are penalized. The new objective function is called Lagrangian function.

The lagrangian problem solution is always lower or equal to the solution of the original problem. (Fisher 2004). The iterative process will bring the solution of the Lagrangian problem nearer to the solution of the original problem.

We will relax the constraints that relies variables belonging to different jobs. Our aim is to divide the complex problem into several easier problems. If we eliminate de restriction of the capacity of the resources, we will be able to split the problem into N job subproblems.

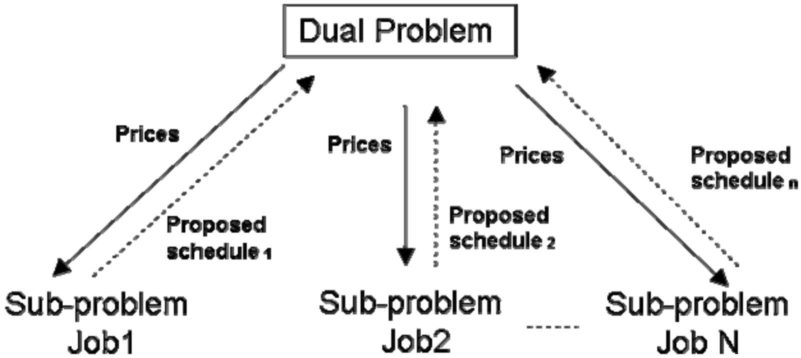


Fig. 3 Problem-solution schema

5 Research Line

We search a method for scheduling manufacturing systems in dynamic environments. In our work we will use a specific problem as Job Shop scheduling problem, but we want to extend the conclusions that we will obtain to other problems of the same nature.

Our objetive is to compare the different methods of updating prices based on those that update the lagrangian problem. (Jihua Wang et al. 1997) One of the most used method is subgradient methods (Geoffrion 1974) (Fisher 2004), (Guignard 2003). There are also other methods derived from the former, e.g. surrogate gradient method (Camerini et al. 1975), (Brännlund 1995), conjugate subgradient method (Jihua Wang et al. 1997) or interleaved surrogate method (Zhao et al. 1997), (Chen & PB Luh 2003). There is no work which compares the updating prices methods of combinatorial auctions for Job Shop problem to the best of our knowledge.

We will define and implement a distributed task scheduling system based on the Job Shop Problem (Pinedo 2008). We want to compare the different methods of updating prices using as criteria convergence, stability and other points to study in a real case implementation as asynchronic computation. We will use different benchmark as we can find in (Demirkol et al. 1998) and the modifications suggested in (Kreipl 2000).

Acknowledgements. This work has been partially supported Caja de Burgos through Projects 2009/00199/001 and 2009/00148/001.

References

Brännlund, U.: A generalized subgradient method with relaxation step. Mathematical Programming 71(2), 207–219 (1995)

Camerini, P.M., Fratta, L., Maffioli, F.: On improving relaxation methods by modified gradient techniques. In: Nondifferentiable Optimization, pp. 26–34 (1975), http://dx.doi.org/10.1007/BFb0120697 (Accedido Junio 3, 2009)

Chen, H., Luh, P.: An alternative framework to Lagrangian relaxation approach for job shop scheduling. European Journal of Operational Research 149(3), 499–512 (2003)

Demirkol, E., Mehta, S., Uzsoy, R.: Benchmarks for shop scheduling problems. European Journal of Operational Research 109(1), 137–141 (1998)

Dewan, P., Joshi, S.: Auction-based distributed scheduling in a dynamic job shop environment. International Journal of Production Research 40(5), 1173–1191 (2002)

Fisher, M.L.: The Lagrangian Relaxation Method for Solving Integer Programming Problems. Management Science 50(suppl. 12), 1861–1871 (2004)

Geoffrion, A.M.: Lagrangean relaxation for integer programming. In: Approaches to Integer Programming, pp. 82–114 (1974), http://dx.doi.org/10.1007/BFb0120690 (Accedido Mayo 20, 2009)

Guignard, M.: Lagrangean relaxation. TOP 11(2), 151–200 (2003)

Kaskavelis, C.A., Caramanis, M.C.: Efficient Lagrangian relaxation algorithms for industry size job-shop scheduling problems. IIE Transactions 30(11), 1085–1097 (1998)

Kreipl, S.: A large step random walk for minimizing total weighted tardiness in a job shop. Journal of Scheduling 3(3), 125–138 (2000)

Kutanoglu, E., Wu, S.D.: On combinatorial auction and Lagrangean relaxation for distributed resource scheduling. IIE Transactions 31(9), 813–826 (1999)

Lee, J.H., Kim, C.O.: Multi-agent systems applications in manufacturing systems and supply chain management: a review paper. International Journal of Production Research 46(1), 265 (2008)

Ouelhadj, D., Petrovic, S.: A survey of dynamic scheduling in manufacturing systems. Journal of Scheduling 12(4), 417–431 (2009)

Pinedo, E.P.M.L.: Scheduling, 2nd edn. Springer, New York (2008)

Shen, W.: Distributed manufacturing scheduling using intelligent agents. Intelligent Systems 17(1), 94, 88 (2002)

Shen, W., et al.: Applications of agent-based systems in intelligent manufacturing: An updated review. Advanced Engineering Informatics 20(4), 415–431 (2006)

Sun, T., Luh, P., Liu, M.: Lagrangian relaxation for complex job shop scheduling. In: Proceedings of 2006 IEEE International Conference on Robotics and Automation, ICRA 2006, pp. 1432–1437 (2006)

Tang, L., Xuan, H., Liu, J.: A new Lagrangian relaxation algorithm for hybrid flowshop scheduling to minimize total weighted completion time. Computers & Operations Research 33(11), 3344–3359 (2006)

Wang, J., et al.: An optimization-based algorithm for job shop scheduling. SADHANA 22, 241–256 (1997)

Study on Integrated Model of Lean and Agile Supply Chain Based on Multi-DPs

Jiang Mei Xian, Feng Ding Zhong, Fan Jia Jing ,Yan Lian Lian,
and Jin Shou Song

Abstract. Lean Production and Agile Manufacturing are currently two popular methods of production. These two popular methods are based on different levels of supply chain optimization and there is an integrated space between these two popular methods. In this paper, we analyse the lean and agile production coupling point or Decoupling Point (DP) in two types of supply chain integration model based on some parts' or components' production process, under the constraints of time, cost and other factors in the supply chain operation and production process. Finally we carry on critical path analysis on the selection of the lean and agile production coupling point or Decoupling Point (DP), and then achieve the optimal integrated strategy.

Keywords: Lean Production, Agile manufacturing, multi-DPs, Lean-Agile supply chain.

1 Introduction

Lean Production and Agile Manufacturing are currently two popular methods of production, both supply chains are based on different levels of the supply chain optimization, there is an integrated space. In adapting to external environment, Lean Production and Agile Manufacturing focus on objectives and organizational forms of production and other fields with their own advantages, if we can effective integrate these two modes into the same supply chain, then they can learn from each other, the operation efficiency of the supply chain will significantly be improved. In the academic community, there are also a large number of articles studied on this aero, in which Customer Order Decoupling Point (Customer Order Decoupling Point, CODP) is an important decision-making issue on supply chain integration.

Jiang Mei Xian · Feng Ding Zhong · Fan Jia Jing · Yan Lian Lian · Jin Shou Song
Institute of Industrial Engineering and Logistical Engineering
Zhejiang University of Technology
The MOE Key Laboratory of Mechanical Manufacture and Automation, 310014,
Hangzhou, China
e-mail: fdz@zjut.edu.cn

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 209–216.
springerlink.com

Enterprises do lean production on existing inventories based on market forecast, when customer orders coming, begin the operation of agile to meet customer needs based on the raw materials or prefabricated parts in inventory, Thus there is a Customer Order Decoupling Point in the production process. The so-called CODP means the transferring point from prediction-based lean production (push-type supply chain) to Agile manufacturing (pull-type supply chain) responding quickly to customer demand in the production process [3].

CODP divides the entire supply chain into two parts: the up stream's lean part and down stream's agility part. Lean part is driven by the customer order, and agile part is driven by the primary protective reserve capacity to regulate fluctuations in demand. A different model of supply chain can be builded relying on location of the separation point in the chain, while the location setting depends on the end-user's maximum acceptable lead time and the position of demand diversity occurred.

2 Multi-DPs

In actual production process, the production of a product is closer to a production network not linear, the supply chain also becomes a supply network, CODP also may be multiple, that is, multi-DPs. The following description of multi-DPs in the supply network is shown in Figure 1.

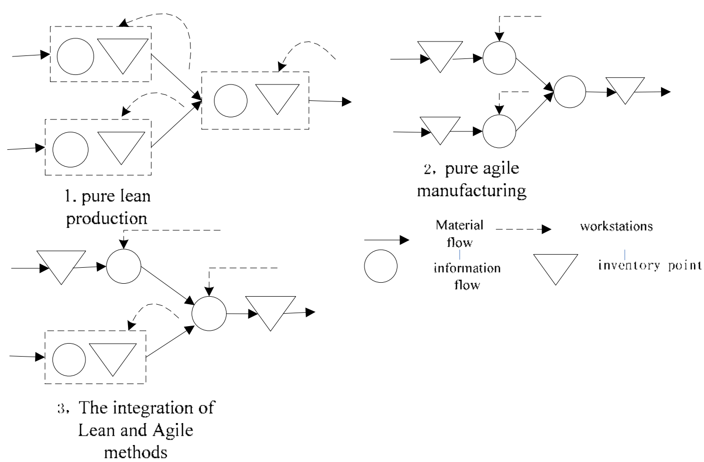


Fig. 1 Multi-DPs

1) Pure lean production (Lean production): all production stages produce on inventory, multi-DPs are at the end of the supply network, while the production lead time is zero.

2) Pure agile manufacturing methods (Agile manufacturing):all production stages produce on order, multi-DPs are at the most front-end of the supply network, while the lead time is the longest.

3) The integration of Lean production and Agile manufacturing methods: in this production mode, the broken line production stages use lean production methods, the remaining two stages use agile methods, while multi-DPs are distributed in different locations of the supply network, the situation is more complex, which is the issues what we want to focus on to study on .

3 Product Structure Tree Model of Core Enterprise in Supply Chain

Figure 2 is a product structure tree of a core enterprise in the supply chain. Fig.2 describes the parts and components needed in the assembly relationship. The necessary components use lean production, if lean production cannot satisfy certain constraints, we should consider to use agile operation. In the following paper LP represents Lean Production, AM represents Agile Manufacturing. So in this supply network, there is an integrated decision-making as to which parts to use lean production, and which parts to use agile operation. To identify multi-DPs is the core of the problem. Fig.2 can not only be used to describe the relationship between the core enterprise and its parts suppliers, manufacturers, and also can represent the function structure model of the entire supply network.

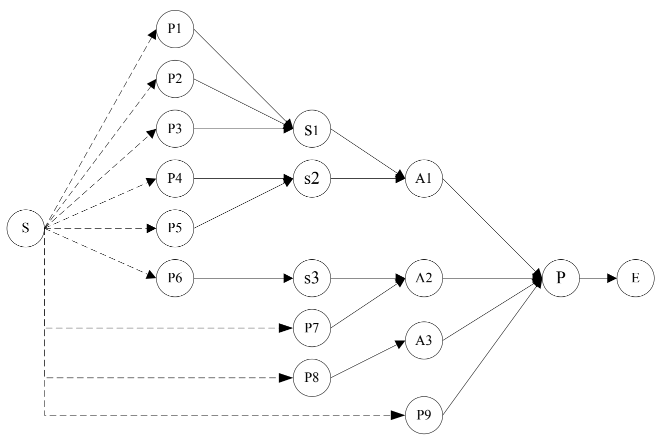


Fig. 2 Product Structure Tree

In Fig.2, P_1 to P_9 indicates the third level part or material, S_1 to S_3 indicates the second level subassembly, A_1 to A_3 indicates the first level assembly, P indicates the final product.

In the paper we use critical path method and oriented graph to describe the product structure tree. Each raw material or part, component and final product is shown as node. We can see there are 16 nodes (the source point S and the end point E do not present physical parts) in Fig.2. So we can get the decision variable $\vec{X}=\left[x_{1,}x_{2,}...x_{15,}x_{16}\right], x_i \in [0,1]$. The arrow line in Fig.2 represents the lead time of production between two nodes, the arrow line from node *i* means the lead time of part *i*. The assembly sequence of parts is shown in Fig.3, the node *i* is assembled from all nodes whose arrow line points to node *i*. Fig.2 shows a oriented graph with nine paths, each path and activity has the former and after tight relationship [6-7].

4 Location Model of Lean and Agile Supply Chain Based on Multi-DPs

Basic symbols are described as follows:

1) *i* means part index, $i \in (1,2.......n)$.

2) Q_i means demand of Q in period i, Q_i submits to $N \sim (\mu_i, \sigma_i^2)$.

3) BOM(*i,j*) means quantity of part i consumed by part *j*, so $\mu_i = \mu_j \times BOM(i,j)$, $\sigma_i^2 = \sigma_j^2 \times BOM(i,j)$.

4) $\vec{X}$ is a binary vector, represents the decision variable of LP/AM, $\vec{X}=\left[x_{1,}x_{2,}...x_{i,}x_n\right], x_i \in [0,1]$, when $x_i = 0$ means part *i* adopts AM, $x_i = 0$ means part *i* adopts LP, the decision variables are bound by the terms of the follow-up decisions.

5) ED_i means expectation demand of part i, $ED_i = \mu_i + z\sqrt{L_i}\sigma_i$, L_i means lead time for purchasing or manufacturing or delivering of part i, Z is a safety coefficient subject to service level α, $Z = [1/2\sqrt{(2/\pi)}]\ln[\alpha/(1-\alpha)]$ [9].

Costs:

1) S_i--- Presetting Production Cost

S_i represents presetting production cost of component *i*, including the cost of order, manufacturing and delivery. S_i is used to measure economy scale which is represented by K_i. When producing according to AM, S_i= K_i, When adopting LP strategy, S_i= K_i/OI_i; here OI_i stands for order interval. The formula $OI_i = \sqrt{2K_i / b_i \mu_i}$, in which b_i stands for the unit holding cost of components.

2) I_i--- Expecting Stock Cost.

I_i represents expecting stock cost of component *i*. $I_i = b_i \int_0^{ED_i} (ED_i - q_i) f(q) dq$ in which "*q*" stands for component demand and *f*(*q*) is the function of its demand probability distribution.

3) Q_i --- Out-of-stock Cost.

Q_i stands for out-of-stock cost of component *i*. $Q_i = b_i \int_{ED_i}^{+\infty} (q_i - ED_i) f(q) dq (\gamma \times c + (1-\gamma) \times e)$. It will happen in two conditions. One is to deal with the problem by treating order as "unfinished" ones, and ratio of this condition is γ. The other is to lose the order all together, which accounts for $1-\gamma$.

Supposing the extra fee of these two conditions is "c" and "e" per unit. Both expecting stock cost and out-of-stock cost are generated only in LP, because AM is formed at the time of receiving orders, which won't lead to expecting stock cost or out-of-stock cost.

4) A_i--- Product Value Cost

A_i represents product value cost of component *i*. $A_i = O_i(\sigma_i/\mu_i)$. O_i stands for the variable cost of component *i*.σ_i/μ_i is defined as variability of demands. A_i is a positive number while adopting LP; and when σ_i/μ_i is relatively large, the cost for component *i* to be used by other products is very high. While adopting AM, A_i=0, because AM begins producing at the time of receiving orders, which won't cause any risk to component *i*.

5) T_i--- Time Cost

T_i stands for the time cost of component *i*. $T_i = O_i\ [(1+\alpha)^{ti}-1]$ [4]. T_i, which merely exists in LP, is mainly decided by the variable product cost . If there is no time restriction, the variable cost of AM and LP is equal to each other. LP needs the holding of stock. Here "t" represents the average stock turnaround time of each component, and α refers to the ratio between capital and time cost.

6) U_i --- Product Updating Cost

U_i stands for product updating cost. $U_i = O_i\ (1- m_i)(1- Rt)/l$ ($Rt<l<(R+1)$ $R\in Z$ [11]. m_i stands for the value ratio of the standard product parts, l is the product life circle, t is the average stock turnaround time.

Table 1 Comparison of Costs

Signs	Costs	AM	LP
S_i	**Preset Production Cost**	K_i	$S_i = K_i / OI_i$
I_i	**Expecting Stock Cost**	**0**	$I_i = b_i \int_0^{ED_i} (ED_i - q_i) f(q) dq$
O_i	**Out-of-stock Cost**	**0**	$O_i = b_i \int_{ED_i}^{+\infty} (q_i - ED_i) f(q) dq (\gamma \times c + (1-\gamma) \times e)$
A_i	**Product Value Cost**	**0**	$A_i = o_i(\sigma_i / \mu_i)$
T_i	**Time Cost**	**0**	$T_i = o_i[(1+\alpha)^{t_i} - 1]$
U_i	**Product Updating Cost**	**0**	$U_i = o_i(1-m_i) * (l - Rt) / l$

Constraints:

PT*i* expresses the production lead time of raw materials or parts, each raw material or component has the production lead time.

Delivery Time is the main Constraints of location model on multi-DPs, $PT(\overline{X}) \le DT$, $PT(\overline{X})$ expresses the lend time of whole supply network, $PT(\overline{X}) = \sum PT_i \bullet x_i$

Objective Function:

Here we take supply chain total cost as the objective function, supply chain total cost (SCTC) is as follows:

$$SCTC = \sum_{i=1}^{n}(S_i + I_i + O_i + A_i + T_i + U_i) = \sum_{i=1}^{n} TC_i^{LP} + \sum_{i=1}^{n}(TC_i^{AM} - TC_i^{LP})x_i \tag{1}$$

Constraints:

$$PT(\overline{X}) = \sum_{1}^{n} PT_i \bullet x_i \le DT, x_i \in \{0,1\} \tag{2}$$

The lead time of purchasing various raw materials, the lead time of product or assemble PT_i *are* known, the constraint is a linear equation, the problem is a 0-1 integer programming model, we can use LINGO or MATLAB software to solve.

5 Analysis of Location Model of Lean and Agile Supply Chain Based on Multi-DPs

1) Analysis of Production Lead Time PTi .

When a part's PT_i decreases, it is easier to meet the need of customer's delivery date, then we can take AM way to produce more parts, the total cost decreases; when a part's PT_i increases, it is more difficult to meet the need of customer's delivery date, then more and more parts are produced in LP way, the total cost increases.

2)Variable Cost Analysis

We can see from the preceding cost description, product value cost A_i, time cost T_i, product updating cost T_i are all formed products' variable cost O_i under different elements' expression.

The time cost $T_i = O_i\ [(1+\alpha)^{ti}-1]$, α means capital time cost rate, t is each component's average inventory turnaround time.

Product updating costs Ui = Oi (1- mi)* (1- Rt)/l, mi is the value rate of standard products, l is product life cycle, t is average inventory turnaround time.

3) Analysis of Changes in Demand

We suppose again that demand amount *f*(*q*) is subject to normal distribution N~$(100,0^2)$, N~$(100,5^2)$and N~$(100,20^2)$.Compared it with N~$(100,10^2)$, we can conclude demand variability's influential trend towards multi-DPs.

Table 2 Analysis of changes in demand

Demand variability	N~(100,02)	N~(100,52)	N~(100,102)	N~(100,202)
The objective	SCTC=75.9	SCTC=210.1	SCTC=285.2	SCTC=345.1
function value	$\vec{X}$	$\vec{X}$	$\vec{X}$	$\vec{X}$
Decision Value	0000000000000000	0000000010000010	0000011111111111	0000011111111111

From Table 2. we know that with the increase of demand uncertainty the whole cost of supply network increases and more parts are produced by AM way.

4) Critical Path Analysis

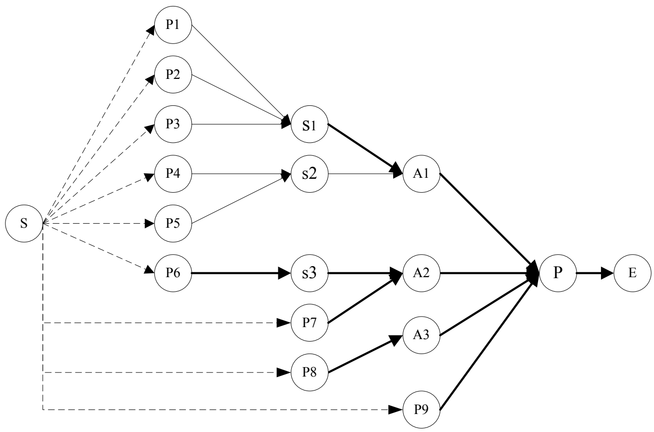


Fig. 3 Critical Path Analysis

According to Figure 3, the longest critical path is P_7-A_2-P–E. It determines the final delivery time $PT(\vec{X}) = 30$ to satisfy customer delivery.

Therefore, if the nodes on the longest critical path are the suppliers, we should take them as strategic supplier. If the nodes are manufacturing activities, we should improve the process and efficiency or productivity. On the other side, if the nodes on the longest non-critical path are suppliers, they needn't supply the product on time. Vice versa, they can longer the time for lower cost. If they are manufacturing activities, we should reform them and pursue lower cost and higher efficiency. This is the essence of critical path analysis.

6 Summary

On the basis of analysis the concept of Customer Order Decoupling Point (CODP), supply chain is actually a net-like supply network with multi-DPs. Based on the product structure tree, this paper describes the research problem and establishes an integrated model of lean and agile supply chain based on multi-DPs. After analyzing the location model of lean and agile supply chain based on multi-DPs, the paper discovers that the integration of lean and agile supply chain is affected by such multiple elements as delivery deadline and demand variability. Enterprises should take the features of their own products and the characteristics of the market into consideration in the integration process, so as to achieve an optimal integrated strategy.

Acknowledgements. This work was supported by Zhejiang Provincial Nature Science Foundation (grant No. Y6090533 and Y6090482) and Zhejiang Provincial Society Science League (grant No. 07N20) and the third "Chief and Senior Professors" support development projects of Zhejiang University of Technology and the MOE Key Laboratory of Mechanical Manufacture and Automation in Zhejiang University of Technology.

References

1. James, P.W., Daniel, T.J.: Lean Thinking: Banish waste and create wealth in your corporation. Simon and Schuster UK Ltd., London (1996)
2. Christopher, M., Denis, R.T.: An integrated model for the design of agile supply chains. International Journal of Physical Distribution& Logistics Management 31(4), 235–246 (2001)
3. Naylor, J.B., Mohamed, M.N., Berry, D.: Leagility: integrating the lean and agile manufacturing paradigms in the total supply chain. Production Economics 62, 10–72 (1999)
4. Feng, D.Z., Chen, L.L., Jiang, M.X.: A Novel Game-theory-based Analysis Approach for Running a Supply Chain Project. In: Huang, D.-S., Li, K., Irwin, G.W. (eds.) ICIC 2006. LNCS (LNAI), vol. 4114, pp. 1014–1019. Springer, Heidelberg (2006)
5. Feng, D.Z., Yamashiro, M.: Optimal Production Policy for a Manufacturing System with Volume Flexibility in a Supply Chain under Lumpy Demand. International Journal of Advanced Manufacturing Technology 25(7-8), 777–784 (2005)
6. Rachel, M.J., Ben, J.N., Denis, R.T.: Engineering the leagile supply chain. International Journal of Agile Management Systems 2(1), 54–61 (2000)
7. Sun, X.Y., Ji, P., Sun, L.Y.: Positioning multiple decoupling points in a supply network. International Journal of Production Economics 113, 943–956 (2008)
8. Ying, L.W.: System engineering theory, method and application. Higher education press, Beijing (2001)
9. Hsu, H.M., Wang, W.P.: Dynamic programming for delayed product differentiation. European Journal of Operational Research 156, 183–193 (2004)
10. Soman, C.A., Donk, D.P., Gaalman, G.J.C.: Combined make-to-order and make-to-stock in a food production system. International Journal of Production Economics 90, 223–235 (2004)
11. Wang, S., Shaligram, P., Wang, Y.L.: Supply chain positioning strategy integration, evaluation, simulation, and optimization. Computers & Industrial Engineering 46, 781–792 (2004)

Towards Improving Supply Chain Coordination through Agent-Based Simulation

Areti Manataki, Yun-Heh Chen-Burger, and Michael Rovatsos

Abstract. One of the most significant paradigm shifts of modern business management is that individual businesses no longer compete as autonomous entities but rather as supply chains. However, the majority of companies, especially small and medium enterprises, fail to design and manage their supply chains in a profitable way, as it is difficult to understand the complex dynamics of Supply Chain Management (SCM). In this paper we argue that agent technologies can provide an intelligent solution to the improvement of SCM. We present a multiagent-based framework for simulating supply chain (SC) operation and re-configuration, with the vision of helping to improve overall SC performance and coordination. The suggested key innovation lies in the better explanation of simulation results and its attractiveness to SCM practitioners. Its theoretical conceptualisation, a logic-based formalisation and the system's architecture that combines agent technologies with business rules and business process modelling are presented.

1 Introduction

Since the 1990s firms realised the importance of Supply Chain Management for gaining and sustaining a competitive advantage. It is only recently, however, that the business world has moved from the traditional "competing firms" model to a "competing supply networks" perspective [12, 8]. The current vision of SCM involves the alignment and integration of SCs, hence a systemic and holistic view of SCM is becoming prominent.

However, SC reality is far from this vision, as most SCs are underperforming and no integration is achieved. This is due to the high complexity of SCM and the lack of understanding of SCM dynamics, especially when it comes to overall SC coordination [13].

Existing approaches towards this problem span over two extremes: they are either theoretical, thus not dealing with SCM dynamics; or they are computational, but with limited theoretical grounding. Furthermore, the few simulation tools that

Areti Manataki · Yun-Heh Chen-Burger · Michael Rovatsos
School of Informatics, The University of Edinburgh, Edinburgh EH8 9AB, UK
e-mail: A.Manataki@sms.ed.ac.uk,
{jessicac,mrovatso}@inf.ed.ac.uk

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 217–224.
springerlink.com

tackle the SC coordination problem are not useful in the new era of SC-based competition as they do not adopt a holistic view of SCM, and they fail to explain the business rationale behind simulation results. On the other hand, agent technologies have been successfully applied for improving SCM [11], but limited work has been done on SC coordination. Moreover, despite their success, multiagent systems are not widely adopted in industry.

Considering the deficiencies of existing approaches, we have identified the following requirements for a solution to the SC coordination problem: An intelligent solution is needed [10] that focuses on inter-organisational coordination [13]. The ability to explain simulation results and attract SC managers are also important. The latter means that the solution should be easy to use and use the same language as SCM practitioners.

In this paper it is argued that agent technologies can provide an intelligent solution for SCM improvement that satisfies all the above-mentioned requirements. We thus present a three-tiered multiagent-based framework for simulating SC operation and re-configuration, with the vision to help improve overall SC performance and enhance SC coordination, thus assisting SCM decision-making. The intelligence provided by agent technologies and the focus on global aspects of SCM (i.e. overall SC performance and SC-wide coordination) tackle the first two requirements. Furthermore, a logic-based approach is suggested in order to enable better explainability of simulation results. Finally, we argue that a theoretical grounding, the adoption of SCM standards and the incorporation of technologies that are popular in industry, will make the suggested solution appealing to SCM practitioners.

The remainder of the paper is as follows: Section 2 provides background information on SCM, explains the motivations for the use of agent technologies for SCM, and describes relevant work. Section 3 presents the suggested three-phased multiagent-based framework for SC simulation, and the system architecture and design decisions are explained. Section 4 discusses lessons learnt and future work.

2 Background Information

A *Supply Chain* is defined as "all parties involved, directly or indirectly, in fulfilling a customer request" [3]. Along this network of parties (e.g. manufacturers, suppliers, distributors, retailers, customers, etc.) there are three types of flows: downstream flow of products, upstream flow of funds and a bidirectional flow of information. Managing these flows with the objective of maximising total SC profitability is defined as Supply Chain Management. Managing a SC is a difficult task due to its dynamic nature and the SC members' conflicting objectives. We identify three main SCM problems: SC planning and demand forecasting provides estimations on future demand, SC configuration specifies the system's structure, policies and processes in a static way, while SC operation/coordination refers to the SC members' actions and interactions, leading to the above-mentioned flows.

Multiagent Systems (MAS) have been widely applied on the SCM context, as they match the SC nature. As Moyaux et al. [11] argue, "supply chains are made up of heterogeneous production subsystems gathered in vast dynamic and virtual coalitions; intelligent distributed systems, e.g. MAS, enable increased autonomy of each member in the SC". SC members also have the same characteristics as agents [11]: They are autonomous (i.e. the business operation of a SC member does not involve direct intervention of others, while it has control over its actions and internal state), social (i.e. there is high interaction between SC members), pro-active (i.e. they perceive their environment, especially their market and competition, and respond to it), and reactive (i.e. they take initiatives in order to maximise their profits). Further motivations for using MAS for SCM are provided in [10], including distributed problem solving and the facilitation of SC integration. It is also worth mentioning that agent technologies are suitable for tackling all three main SCM problems, either by automating, simulating or recommending solutions. Agent-based negotiation and optimisation can be utilised for SC configuration, agent-based coordination is linked to SC operation, while the agents' learning capabilities can facilitate SC planning.

Representative work in agent-based simulation for SC coordination and configuration includes [5] and [15]. The former adopts a systemic view of SCM and introduces conversation plans that capture coordination knowledge; however no supportive business theories are presented and SC operations are not explicitly addressed. The latter utilises a library of structural and control elements and a classification of messages to simulating all SC flows, but it does not support knowledge discovery or explain simulation results. SC configuration is automated in [4] through the combination of MAS with machine learning; however, no underlying business theory is presented.

3 A Multiagent-Based Framework for Overall SC Simulation

With the aim of providing an intelligent solution for SCM improvement, that is realistic and easy to use by industry, we propose a three-phased agent-based SC modelling and simulation framework. This framework tackles the SC operation and configuration problems, and complies with business theories and SCM standards. As shown in Fig. 1, the suggested framework consists of three phases: conceptualisation, formalisation and simulation of SCM. The *conceptualisation* phase involves abstracting the domain of SCM through suitable constructs. During the *formalisation* phase the selected constructs of Phase 1 are defined and libraries of SCM concepts are developed. The *simulation* phase utilises the libraries of Phase 2 and simulates SC operation and configuration (i.e. SC partner selection), thus explaining dynamic aspects of SCM and allowing for experimentation. Note that throughout our framework a logic-based approach is adopted in order to allow for knowledge-enriched analysis of the domain.

3.1 Conceptualising SCM

We conceptualise SCM through three basic components: SC roles, SC services and SCM processes. Each SC role provides a specific SC service through the execution of the corresponding enabling SCM processes. Hence, we regard the SC as

Fig. 1 Three-phased framework for SC simulation

a virtual organisation (VO) with the goal of satisfying the requirements of the final customer and maximising total SC performance. A high-level plan towards achieving this goal consists of basic functions or tasks, which we call SC services. SC members may bid for and be delegated a specific SC service, thus being assigned the corresponding SC role. Last, a SC role prescribes the execution of SCM processes that realise the delegated SC service; hence, the SCM process model of a SC role constitutes its internal plan towards delivering the delegated SC service.

A *SC service* is defined as an archetypical function that supports the flow of products requested by the final customer. Illustrative examples include the following: manufacture final product, sell product to final customer, transport product, etc. A *SC role* is defined as the combination of the archetypical service of a SC member towards the SC (and hence the corresponding position in the SC network) and its business model, i.e. how it makes money from what it does [17]. Adopting the business model typology by [17], we recognise SC roles such as supplier-creator, manufacturer and retailer. Note that a SC member may adopt multiple SC roles within a certain SC. Last, a *SCM process* is defined as a SC role's business process supporting the flow of the requested product to the final customer. Receive product, send invoice and consolidate order are examples of SCM processes.

In order to make our solution easier to understand and more realistic and attractive to SC managers, we have based our conceptualisation on widely accepted business theory. Hence, adopting a holistic approach to SCM, we view the SC as a VO, where the theory of organization design can be applied. Galbraith [6] has suggested a star model of the five major components of organization design, among which processes and structure are the prominent ones. The main dimensions of organizational structure are power and authority, reporting relationships, and organizational roles. But since SCs are known to have a chain VO structure with respect to power, authority and reporting relationships, it is reasonable to minimise SC structure to the SC role dimension. Furthermore, because SC roles are based on archetypical SC services, SC services are a construct of our conceptual model. Lastly, by recognising SCM processes as a main component of SCM, we comply with [1], according to which process improvement orientation is the most popular SCM construct.

3.2 Formalising SCM

Adopting a logic-based approach, we use first-order predicate logic to define the SCM constructs of Phase 1, as shown at the schemata provided below. A SC service is defined through its id, name and position in the SC (i.e. upstream or downstream). A SC role is defined through its id, name, the provided SC service, the corresponding business model and its SC position. The schema for SCM process complies with the Fundamental Business Process Modelling Language (FBPML), a logic-based business process modelling language that guarantees both rich visual modelling methods and formal semantics [2]. Following the FBPML activity specification, a SCM process is defined through its hierarchical position (id), name, triggering events, preconditions and actions.

```
SCservice(ID, Name, Position)
SCrole(ID, Name, Service, BusinessModel, NetworkPosition)
SCMprocess(Position, Name, TriggeringEvents, Preconditions, Actions)
```

The formalisation phase also involves populating libraries for the SCM constructs, thus providing instances that can later serve as simulation building blocks. In order to make these libraries SCM practitioner-friendly, they should be based on widely accepted SCM theory. Therefore, we suggest the use of the Supply Chain Operations Reference (SCOR) model [14] for the SCM process library, as it is the most widely used standard within the SCM community.

3.3 Simulating SCM

This phase simulates SCM behaviours with respect to the SC operation and configuration problems, and hence two interrelated modules are recognised. The SC operation module simulates the actions and interactions of SC members for a given SC configuration and measures overall SC performance. The SC configuration module simulates the selection and contracting process with SC partners for any SC member. Hence, the integration of the two modules allows simulation of the SC operation where the SC network may be reconfigured at run-time. This way, complicated questions can be answered, such as: How flexible is the SC in the case where supplier X fails to deliver a promised order? In this paper we focus on the SC operation module, while a brief introduction is given on the SC configuration module.

An agent-based *conceptual design* of the simulation environment is proposed. We regard each SC member as an intelligent agent that can decide on SCM issues, execute processes and communicate with other SC members. Simply put, in our framework a SC member can think, act and interact, and is consequently conceptualised as an agent consisting of three layers:

- Reasoning layer: corresponds to the beliefs, desires and intentions of the agent, and drives its decision-making towards actions
- Process layer: corresponds to the agent's ability to execute processes, thus acting upon the environment
- Communication layer: corresponds to the agent's ability to receive and send messages to other agents.

We suggest the use of business rules (BR) to represent a SC agent's reasoning layer, while its decision-making process can be driven through a reasoning engine. The SC agent's process layer can be represented by a business process model (BPM), the execution of which will be facilitated with the use of a workflow engine. Finally, the SC agent's communication layer can be represented by communication predicates, which will operate upon a communication environment.

3.3.1 Functionality and Architecture of the SC Operation Module

The SC operation module reads a SC configuration and simulates the actions and interactions between SC members. Specifically, its inputs include information on SC members, market demand and the final product. Its outputs include real-time SC operation information (e.g. supplier X has completed the execution of process p34-receive order), overall SC performance (e.g. overall cost is 325), and business process analysis results (e.g. possible bottleneck p36 for supplier s12).

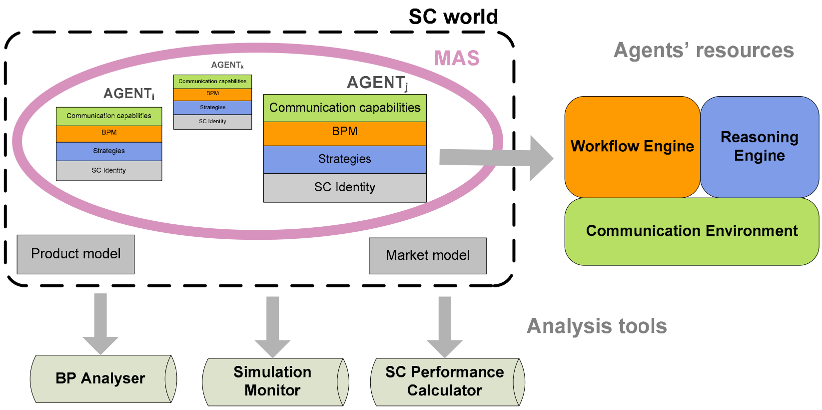


Fig. 2 Architecture of the SC operation simulation module

The suggested architecture for the SC operation simulation module is presented in Fig. 2 and it is based on the functionality and conceptual design explained previously. Three main components can be seen in Fig. 2: SC world, agents' resources and analysis tools. The *SC world* consists of a MAS of SC agents, the SC product and the market demand for that product. The product and market models are static models describing the SC's product (e.g. bill of materials, attributes, etc.) and the final market demand (e.g. average order amount and order frequency). A SC agent consists of four subcomponents: SC identity, strategies, BPM and communication capabilities, which define the SC agent in a static way. In order to exhibit dynamic behaviour, a SC agent uses resources that drive SC simulation. The *resources* that are available to SC agents are: a workflow engine, a reasoning engine and a communication environment. As implied by the colours in Fig.2, these resources are linked to the SC agent's components: The workflow engine executes processes of an agent's BPM, and thus updates its workflow state. Similarly, the reasoning engine reads the SC agent's strategies (defined as BR) and turns them into decisions towards actions for each state. The communication environment allows the exchange of messages within the SC through an appropriate infrastructure (e.g. message exchange channels or a blackboard) and a protocol. Lastly, the analysis of the simulation is enabled by three *analysis tools*: First, the simulation monitor is understood as a simulation controller, making sure that the SC world is correctly modelled and that no rules are broken during simulation.

Second, the SC performance calculator reads the overall SC simulation results and computes its performance in terms of time and cost. Third, the BP analyser studies the overall SC workflow in order to detect bottlenecks, unreachable points, etc.

Adopting a knowledge-based approach, we suggest a *logic-based implementation* of the SC simulation environment. This is work in progress; however a first version of a declarative workflow engine has been developed and tested on a real-world case. The declarative workflow engine DeWE [9] has been designed with respect to FBPML specification to simulate BPM execution and measure time and cost. DeWE has been successfully used for analysing Dell's SCM, and hence it is a good starting point for driving the process-oriented behaviour of SC agents.

3.3.2 Design Decisions Regarding the SC Configuration Module

The SC configuration module is used when a SC member needs a new partner upstream or downstream, and it simulates the selection and contracting process. Facilitating SC configuration is important, as the reconfiguration process takes place at run-time, thus resulting in possible delays and affecting overall SC performance. Furthermore, a new SC configuration defines new rules of actions and interactions between SC members, thus affecting SC coordination. As far as the module's functionality is concerned, its inputs include the selection criteria of the contracting SC member, the attributes of all candidates and their trading strategies. Its output is the new SC configuration, as defined by the agreed contract.

We propose the use of three agent-based mechanisms for simulating the SC configuration process. Firstly, an auction takes place, which allows the interested SC member to attract possible SC partners and receive offers. Secondly, the SC member selects one of the offers/candidates based on internal criteria (i.e. a constraint satisfaction problem). Thirdly, the SC member may negotiate with the selected candidate to set a contract. In order to make the simulation of this problem as realistic as possible, we suggest the use of appropriate business theories for the interaction protocol and the selection process, such as [7, 16].

4 Conclusions and Future Work

Recognising the recent shift towards SC-based competition, and the need for concepts and tools to assess the performance of an entire SC [12], we have suggested a three-phased multiagent-based framework for conceptualising, formalising and simulating SC operation and configuration. We believe that a theoretically well grounded conceptualisation, a formalisation adopting the SCOR model, and the coupling of agent technologies with business process modelling and business rules for the simulation, will make the system more appealing to SC managers and will allow them to directly incorporate their business rationale. By adopting a logic-based approach, we also wish to make this business rationale transparent and support the user's understanding of the SC simulation results. Furthermore, the combination of agent technologies with business operations, the incorporation of the effect of delays due to real-time communication or SC configuration on overall SC performance, and the provided business process analysis throughout the SC are aspects of the added value of the proposed system.

With the vision of an even more realistic agent-based simulation environment, our future work is to design a SCM communication protocol based on appropriate business theories and complete the implementation of the suggested framework.

References

1. Burgess, K., Singh, P.J., Koroglu, R.: Supply chain management: a structured literature review and implications for future research. International Journal of Operations & Production Management 26, 703–729 (2006)
2. Chen-Burger, Y.H., Tate, A., Robertson, D.: Enterprise Modelling: A Declarative Approach for FBPML. In: European Conference of Artificial Intelligence, ECAI, Lyon, France (2002)
3. Chopra, S., Meindl, P.: Supply Chain Management: Strategy, Planning and Operation. Prentice Hall, Englewood Cliffs (2003)
4. Emerson, D., Piramuthu, S.: Agent-Based Framework for Dynamic Supply Chain Configuration. In: Proceedings of the 37th Annual Hawaii International Conference on System Sciences, January 5-8. IEEE Computer Society, Washington (2004)
5. Fox, M.S., Barbuceanu, M., Teigen, R.: Agent-Oriented Supply Chain Management. International Journal of Flexible Manufacturing Systems 12(2-3), 165–188 (2000)
6. Galbraith, J.: Designing Organizations: An Executive Briefing on Strategy, Structure, and Process. Jossey-Bass, San Francisco (1995)
7. Goldkuhl, G.: The six phases of business processes – business communication and the exchange of value. In: 12th biennial ITS conference Beyond convergence (ITS 1998), Stockholm (1998)
8. Lambert, D.M., Cooper, M.C.: Issues in Supply Chain Management. Industrial Marketing Management 29(1), 65–83 (2000)
9. Manataki, A., Chen-Burger, Y.H.: Analysing Supply Chain Strategies Using Knowledge-Based Techniques. In: Proceedings of the 3rd KES Symposium on Agents and Multi-Agent Systems – Technologies and Applications, Invited Session on Digital Economy, Uppsala, Sweden, June 2009. Springer, Berlin (2009)
10. Mitkas, P.A., Nikolaidou, P.: Agents and Multi-Agent Systems in Supply Chain Management: An Overview. In: Protogeros, N. (ed.) Agent and Web Service Technologies in Virtual Enterprises, ch. 13. Information Science Reference, New York (2008)
11. Moyaux, T., Chaib-draa, B., D'Amours, S.: Supply Chain Management and Multiagent Systems: An Overview. In: Chaib-draa, B., Müller, J.P. (eds.) Multiagent based Supply Chain Management, ch.1. Springer, Berlin (2006)
12. Srai, J.G., Gregory, M.: A supply network configuration perspective on international supply chain development. International Journal of Operations Management 28(5), 386–411 (2008)
13. Stadtler, H., Kilger, C.: Supply chain management and advanced planning: concepts, models, software and case studies. Springer, Heidelberg (2005)
14. Supply Chain Council, Supply-Chain Operations Reference-model, Version 9.0 (2008), http://www.supply-chain.org
15. Swaminathan, J.M., Smith, S.F., Sadeh, M.: Modeling Supply Chain Dynamics: A Multiagent Approach. Decision Sciences 29(3), 607–632 (1998)
16. Weber, C.A., Current, J.R., Benton, W.C.: Vendor selection criteria and methods. European Journal of Operational Research 50, 2–18 (1991)
17. Weill, P., Malone, T.W., D'Urso, V.T., Herman, G., Woerner, S.: Do Some Business Models Perform Better than Others? A Study of the 1000 Largest US Firms, MIT Center for Coordination Science Working Paper No. 226, MIT Sloan School Working Paper 4615-06 (2004)

Using Multi-agent System for Improving and Implementing a New Enterprise Modeling Tool

Paul-Eric Dossou and Pawel Pawlewski

Abstract. GRAI Methodology is one of the three main methodologies for enterprise modeling (with PERA and CIMOSA). To support this methodology different tools are being developed. GRAIXPERT is a hybrid expert system for detecting inconsistencies in enterprises. GRAISUC is a module for choosing and implementing supply chain management tool in enterprises. GRAIQUAL is a module for improving quality system of enterprises. These modules are based on the use of different reasoning (Case-based reasoning (CBR), decomposition, transformation) but also on the enterprise knowledge management. The enterprise knowledge could be explicit or tacit. Each case of enterprise studied allows to improve the knowledge of the tool. Then, Multi-agent systems are used for acquiring this knowledge and managing improvement. This paper presents how Multi-agent system could be associated to CBR and Decomposition reasoning in order to be more efficient during the enterprise modelling.

Keywords: Multi-agent systems, Case-Based Reasoning, Expert system, enterprise modelling, performance, reference models, rules, Knowledge.

1 Introduction

Different methodologies are used for improving enterprise performance. GRAI methodology is one of the three main methodologies with PERA and CIMOSA [11]. GRAI approach is composed of five phases: an initialization phase for acquiring the context of the enterprise studied, the modeling phase for elaborating enterprise models (functional, physical, decisional, process, informational) according to

Paul-Eric Dossou
CAM, Site de Vendée, 28 Boulevard d'Angleterre,
85000 La Roche-Sur-Yon, France
e-mail: paul-eric.dossou@icam.fr

Pawel Pawlewski
Poznan University of Technology, ul.Strzelecka 11
60-965 Poznań
e-mail: pawel.pawlewski@put.poznan.pl

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 225–232.
springerlink.com

different points of view, the diagnosis phase for detecting inconsistencies in the models, design phase for improving the models, and the last one which is the implementation of the new system. Some tools are being developing in order to support this methodology. GRAIXPERT is a hybrid expert system used for managing, rules, reference models and knowledge of specialists. GRAISUC allows to manage the choice and the implementation of a supply chain management tool in enterprise. GRAIQUAL is used for managing the implementation of a global quality approach in the enterprise.

For developing these modules, the necessity of using intelligent CAD systems seems to be evident. The problem solving systems using only one type of reasoning are not efficient. Different reasoning (CBR, decomposition, direct correspondence, transforming reasoning) have to be combined. According to CBR philosophy, the case base needs to contain a significant number of cases in order to be efficient. It means knowledge needs to be continuously updated. The combination of this problem solving method and multi-agent systems seems to be an ideal solution. Indeed, for developing each module, JADE (Java Agent Development Framework) technology which implements multi-system agents with the standard FIPA-ACL language (Foundation for Intelligent Physical Agents – Agents Communication Language) is used.

After presenting the concepts of the modules being developed, the combining of reasoning with multi-agent systems theory will be shown.

2 GRAIXPERT, GRAISUC and GRAIQUAL

GRAI methodological tree shows five enterprise domains and the modules associated to each domain (Fig. 1).

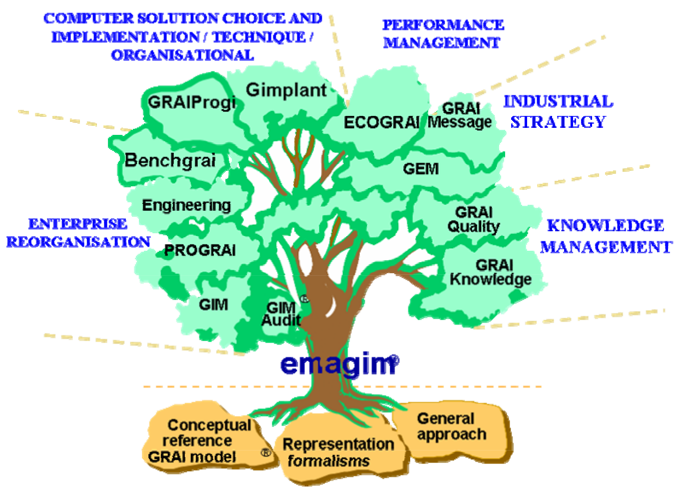


Fig. 1 Rules based reasoning mechanism for the diagnosis

GRAIPROGI is the module for choosing and implementing tools in an enterprise. GIM Audit is the module for modelling enterprises. GRAI Quality is the module for improving quality. These modules are respectively supported by three software tools: GRAISUC, GRAIXPERT and GRAIQUAL. These tools are used for solving the problem of the existing enterprise but also for acquiring knowledge for the next enterprises which will be studied.

The architecture of GRAIXPERT is composed of three sub-modules in interaction with e-MAGIM Kernel (Fig. 2): the Manager, the Knowledge Capitalization and the Knowledge Based System [3], [4],[10] . The Kernel is used for modeling the existing system. It is a graphic editor used for representing various models of the GRAI methodology.

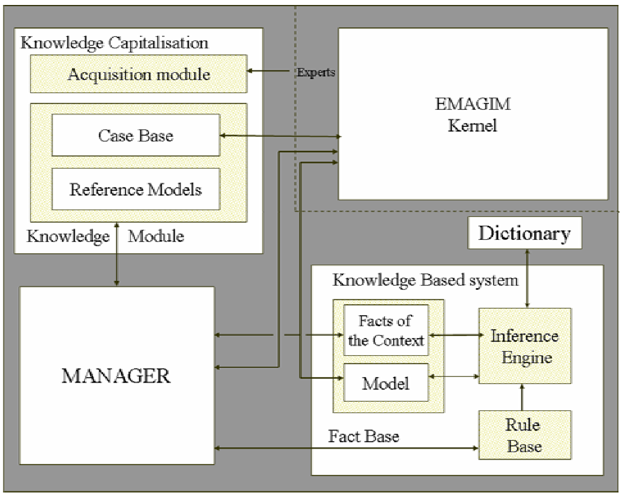


Fig. 2 Architecture of GRAIXPERT

The **Manager** controls and manages the system's interactions with the users. It presents to the users with some appropriated questions and choices, the necessary information about the characteristic of the studied enterprise. It also manages the rules classified according to a typology of production systems. Its main tasks are modifying, suppression, or selection of the applicable rules in a given context. It is also used for the loading and the saving of rules files. Finally, it controls the design process, different actions of the sub-modules and their interactions. The **Knowledge Capitalization** is composed of reference models associated to a domain and of previous studies. The knowledge capitalization process needs some aptitudes to manage different know-how and points of view. It must integrate this knowledge in an accessible, usable and maintainable form. It is proposed an expertise model based on the knowledge of the experts but also on the previously realised studies.

The knowledge capitalization module contains two sub modules:

- The acquisition one is used to improve the acquired knowledge. It serves to modify, add, and suppress the knowledge in the Knowledge sub-module.
- The knowledge sub-module with two databases. The first is the case base where previous studies are stored. The second is composed of reference models [5].

The **Knowledge Based System** contains a rule base (Fig. 3), a fact base to store the existing system's models obtained with e-MAGIM Kernel, but also the fact base of the context containing the data of the context and corresponding to facts used in the rule base, and an inference engine. A dictionary is used to translate user's expressions into standard expressions provided by the GRAI methodology.

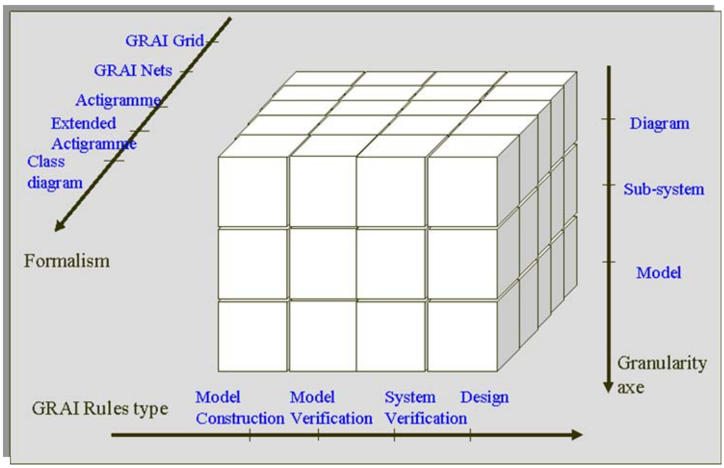


Fig. 3 Structure of the rule base

The architecture of GRAIQUAL and GRAISUC are elaborated with the same concepts than GRAIXPERT. The architecture of GRAIQUAL is composed of a Management Module (MM), a Work Base (WB), a Transfer Interface (TI), an Improvements Management Module (IMM) and a Knowledge Base Module (KBM).

The **Management Module** is used for organising the different interactions of the tool with the expert system (GRAIXPERT) and the kernel of e-Magim. The **Work Base** is being elaborated for managing and capitalizing knowledge about the studied case. It's the space used for improving the enterprise especially the quality system of the enterprise. The **Transfer Interface** is used for putting the new case in GRAIXPERT in order to improve the Case Base. The reference model elaborated for each enterprise domain will be improved by the acquisition of new models in GRAIXPERT. The **Improvements Management Module** is also being developed for managing the different quality action plans of the enterprise. It contains different quality tools. A **Knowledge Base Module** is being elaborated for containing the rules according to quality certifications.

The IMM contains tools like SPC (Statistical Process Control), Poka-Yoke, QFD (Quality Function Deployment), PDCA (Plan, Do, Check, Act), Hoshin, Kanban, 5S and Lean Manufacturing. Norms such as HACCP, BRC or IFS for the agribusiness sector are stored in the KBM. The HACCP (Hazard Analysis Critical Control Point) consists in identifying and evaluating the risks associated with various stages of the production of edible products and to define the necessary means to master them.

The architecture GRAISUC contains a Management Module (MM), a Work Base (WB), a Transfer Interface (TI), Specifications Management Module (SMM) and a Specifications Capitalization Module (SpCM). The cooperation of different modules is possible because of the same data base and the same formalisms [6].

The **Management Module** will be used for organising the different interactions of the tool with the expert system (GRAIXPERT) and the kernel of e-MAGIM. The **Work Base** is being elaborated for managing and capitalizing knowledge about the studied case. The **Transfer Interface** is used for putting the new case in GRAIXPERT in order to improve the Case Base. The reference model elaborated for each enterprise domain will be improved by the acquisition of new model in GRAIXPERT. The **Specification Management Module** is also being developed for obtaining the adapted SCM Tool of the enterprise. The different specifications obtained by analysing enterprises in order to choose SCM tool will be capitalized in the **Specification Capitalisation Module**.

These different architectures are integrated. They are based on a problem solving method combining CBR, decomposition, direct correspondence and transforming reasoning. In this paper, a zoom is made on the sub-systems of the different architectures and show how multi-agent systems could allow to manage the tools.

3 Multi-agent Systems

A state of the art of the agents allows to notice the specificity of each type of agents in order to choose the right one for our problem. The **reactive agents** react only for the changes of the environment, and don't include reasoning for problem solving. The Deliberative agents make a certain deliberation for choosing their actions. The BDI architecture (Belief, Desire, Intentions) is an approach used for the design of deliberative agents (Fig. 4). A set of beliefs represents information of the agent about its environment. A beliefs revision function takes the sensors entrances and actual beliefs of the agent and determines the new set of beliefs [9]. The option generation function determines the available options for the agent (i.e. desires) by using actual beliefs on the agent about the environment and the actual intentions. A set of desires represents the options available to the agent. A filter function represents the deliberation process of the agent and determines the intentions of the agent by using its desires, its beliefs, and actual intentions. A set of actual intentions, represents the actual intention centre of the agent. A selection function of actions determines the action to do by using the actual intentions of the agent.

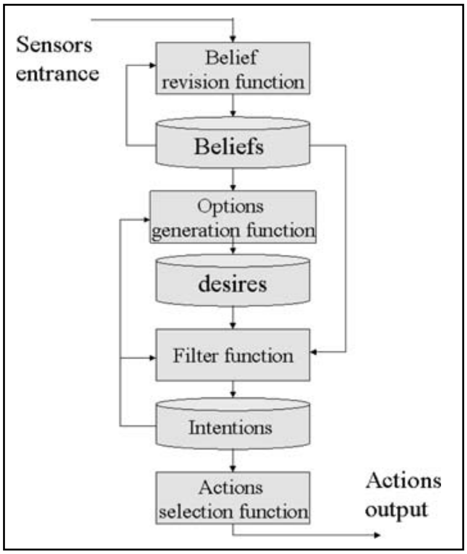


Fig. 4 The BDI architecture

The training agent is composed of 4 parts. The first part is the critical which tells to the training module how the action of the agent is good (Fig. 5). The training module uses a certain retroaction on the actions of the agent for determining how the performance module could be change for improving the system in the future [7], [8]. The performance module is seen as the global agent when there is no training.

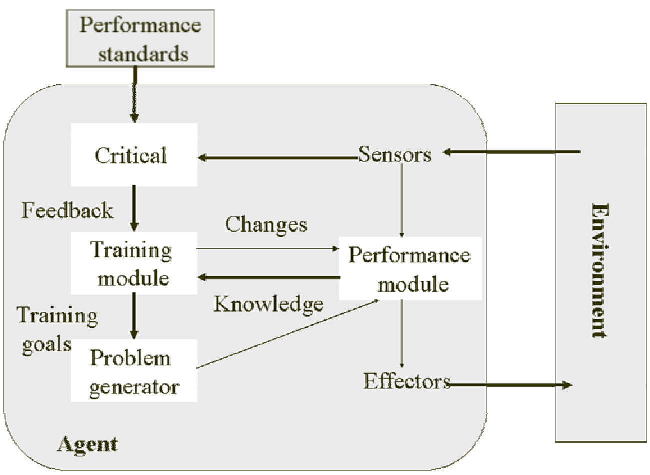


Fig. 5 General model of training agent

The problems generation gives some suggestions of actions for inciting the agent to realize an exploration. This could allow to choose some non-optimised solutions in short term but that could be the best in long term.

4 Combining CBR and Multi-agent Systems

The problem solving method elaborated in the frame of the development of GRAIXPERT, GRAISUC and GRAIQUAL combines different reasoning.

For example, GRAIXPERT is used for modelling enterprises. During the diagnosis phase of GRAI approach, the manager uses the rules contained in the knowledge base. It corresponds to reactive tasks. Then some inconsistencies are obtained. For the design phase, some reference models are established according to the five models of the GRAI methodology (functional, process, physical, informational, and decisional). The Manager allows to choose the good reference model. This step of adaptation contains the preliminary design phase. The first two steps of the CBR algorithm « select- extract », allow to obtain the reference model. We have interaction with the Knowledge Capitalization module. The objective is to compare the existing study to the corresponding reference model. It corresponds to deliberative tasks. For a study including the five models of the enterprise, the decomposition reasoning is applied in order to design each model separately. This reasoning is combined with the CBR (Case-Based Reasoning) [1] [2], and the transformation reasoning. The adaptor is used from the chosen reference model and the associated context for obtaining the preliminary model by using the decomposition reasoning mechanism and the direct correspondence reasoning.

Indeed, each reference model is considered as a structure in its own right. The new model obtained is saved in a case base, but also allows to improve the reference model. It corresponds to training tasks. It means that the reference model will change with experience. The knowledge capitalization module has to be able to learn, and the manager has to be able to integrate new changes of the knowledge module. The acquisition module of the Knowledge module takes into account expert remarks, in order for updating. So the training capacity of these modules is essential to be efficient. The rules contained in the Knowledge Based System could be completed by the experts, but also by the experience of a new analysis. Then the use of training agent appears as the right thing to do.

The Manager of GRAIXPERT is of course directly in interaction with the modules of GRAISUC and GRAIQUAL. Indeed, the Specification Capitalization Module, the Specification Management Module, the Improvement Management Module of GRAISUC and GRAIQUAL represent also multi-agent systems and are building with the same formalisms.

The elaboration of the training agent structure for each of these different modules is being developed in the FIPA-ACL language in order to be in coherence with the JADE platform chosen for developing GRAIXPERT, GRAIQUAL and GRAISUC.

5 Conclusions

The new crisis and globalization that enterprises have to be well-organised in order to optimise their performance on the market and prepare themselves against constraints and urgent events. GRAI Methodology participates for attaining this objective. For supporting expert in the use of this methodology and increasing the audit probability of success, some tools are being developed. These tools use different reasoning, and concepts like CBR and Multi-agent systems. A general process of problem solving is established. The main principles which have to be respected by the problem solving method are shown. Then, according to this general process different reasoning mechanisms are implemented. These mechanisms of reasoning are related to the steps of the enterprise improvement process. A problem solving method associated to the hybrid expert system developed is elaborated. Then, the combining of these reasoning with multi-agent system concepts in order to develop coherent and efficient tools is shown. These tools are being developed by using JADE platform with FIPA-ACL language.

References

1. Aamodt, A.: Case-Based Reasoning: foundational issues, methodological variations, and system approaches. Artificial Intelligence Communications 7(1), 39–59 (1994)
2. Burke, E.K., et al.: Structured cases in case-based reasoning – reusing and adapting cases for time-tabling problems. The journal of KBS 13(2-3), 159–165 (2000)
3. Brown, D.C., Chandrasekaran, B.: Expert system for a class of mechanical design activities. In: Knowledge Engineering in CAD. Elsevier, Amsterdam (1985)
4. Dossou, P.E., Mitchell, P.: Using case based reasoning in GRAIXPERT. In: FAIM 2006, Limerick, Ireland (2006)
5. Dossou, P.E., Mitchell, P.: Implication of Reasoning in GRAIXPERT for modeling Enterprises. In: DCAI 2009, Salamanca, Spain (2009)
6. Dossou, P.E., Mitchell, P.: How Quality Management could improve the Supply Chain performance of SMES. In: FAIM 2009, Middlesbrough, United Kingdom (2009)
7. Russell, S.J., Norvig, P.: Artificial Intelligence. A Modern Approach. Prentice-Hall, Englewood Cliffs (1995)
8. Sen, S., Weiss, G.: Learning in Multiagent Systems. In: Weiss, G. (ed.) Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence, ch. 6, pp. 259–298. The MIT Press, Cambridge (1999)
9. Wooldridge, M.: Intelligent Agents. In: Weiss, G. (ed.) Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence, ch. 1, pp. 27–77. The MIT Press, Cambridge (1999)
10. Xia, Q., et al.: Knowledge architecture and system design for intelligent operation support systems. The Journal Expert Systems with Applications 17(2), 115–127 (1999)
11. Chen, D., Doumeingts, G., Vernadat, F.B.: Architectures for enterprise integration and interoperability. Past, present and future. Computers in Industry 59, 647–659 (2008)

Production Process Based on CIMOSA Modeling Approach and Software Agents

Pawel Pawlewski and Arkadiusz Kawa

Abstract. The paper presents the results of the authors' studies which consisted in modeling the processes taking place in an enterprise characterized by manufacturing of complex products (machine building) of a very long production cycle. The production environment of a diesel ship engine is described. The idea of the so-called domains of the CIMOSA concept which has been used for the modeling was explained. The article introduces the multi-agent system idea on the basis of which a model will be built and used in the simulation experiment. Finally the algorithm for planning process is presented.

Keywords: Distributed manufacturing system, Multi-agent systems, Distributed process planning and scheduling, Process modeling.

1 Introduction

In order to maintain their competitive advantage in the changing world economy enterprises must have the ability of both developing new products quickly and adjusting their production capabilities and their functionality to the current market needs. Enterprises which make highly complex products (machine-building, automotive, aerospace industries) make use of the possibilities offered by the conception of Product Lifecycle Management (PLM) in order to produce them competitively. The PLM approach offers methods of product management and development, information related to the product, including its development, production, marketing, order and delivery processes, throughout the whole product lifecycle – from the initiation (product idea) to its end [8]. It must be stated that

Pawel Pawlewski
Poznan University of Technology
Strzelecka 11, 60-695 Poznań, Poland
e-mail: pawel.pawlewski@put.poznan.pl

Arkadiusz Kawa
Poznań University of Economics
al. Niepodleglości 10, 60-965 Poznań, Poland
e-mail: arkadiusz.kawa@ue.poznan.pl

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 233–240.
springerlink.com

although the PLM conception helps to manage product-related information, its numerous variants and modifications to a large extent, its application in critical production phases is insufficiently defined/developed. The main reason is the fact that additional levels related to the complexity of production processes are not taken into consideration. Plenty of instances (copies) of the product must be made in the required amount, which means that production processes are, in fact, composed of many individual processes (one for each item produced) carried out simultaneously but in different temporal sequences (figure 1). Each of them may also contain additional elements (variants) due to customization.

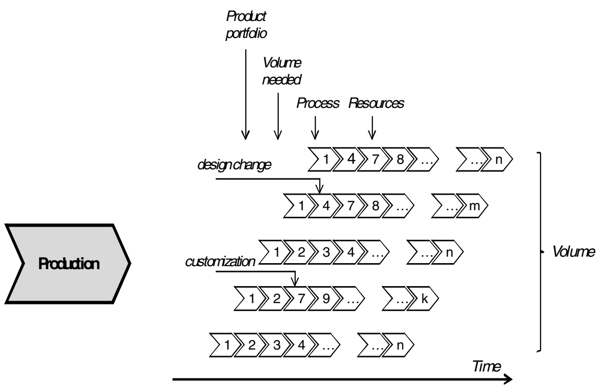


Fig. 1 Complexity of production process and its management [5]

That is why there is an obvious need to expand the basic PLM conception which would allow to take the process perspective – product lifecycle management and the PPLM process (Product and Process Lifecycle Management) – into account fully.

With regard to the need of developing an alternative for the PLM conception – the consideration of the process perspective – the authors have mainly focused on two important phases of the PLM conception – the process planning phase and the production phase. The studies related to these phases published in the world have integrated them into the so-called Enterprise Architecture (EA). It is being predicted that almost 70% of the European Union enterprises will introduce the EA projects in the nearest future [11]. The previous research of the authors has already proven that two approaches are dominating the range of EA-related studies at the moment: one, dominated by computer science, which aims at enterprise integration around information systems and assumes information service of business processes as its staring point, and another one which aims at enterprise integration around productive processes. The worldwide research related to the second approach is mainly based on the CIMOSA and GRAI conceptions [3].

The aim of the paper is to present the results of the authors' studies which consisted in modeling the processes taking place in an enterprise characterized by manufacturing of complex products (machine building) of a very long production cycle. It is unrhythmical production, also characterized by the highest degree of complexity of planning and production controlling problems due to the amount and changeability of labor. The subject of the studies is the production process of ship engines at HCP – a company which is a co-operator in the research conducted by the authors.

The article consists of 4 sections, the second of which presents the production environment of a ship engine at HCP. The third one describes the idea of the so-called Domains of the CIMOSA conception which has been used for the modeling. The fourth section depicts the multi-agent system idea on the basis of which a model will be built and used in the simulation experiment. The fifth section focuses on modeling of long cycle process and the proposals to planning process.

2 The Factory Characteristics

HCP is a factory established in 1857 in Poznań (Poland), and named by its founder Hipolit Cegielski. Nowadays it is the biggest ship engines producer in Europe. HCP produces slow speed-rotation engines for transport ships. The ship engines are built to special orders of customers. Even two engines of same type may have some differences depending on a customer's wishes. HCP builds about 25 – 35 engines per year.

The engines are built under the license of Sulzer Brothers (Wartsila) and Burmeister & Wain. The dimensions of such engines are impressive: over 4 meters wide, over 20 meter long, almost 16 meters high.

A ship engine is a product of a very high capital intensity, and that is why financing its production must be supported by guarantees and bank loans. Therefore, in the company there are two sale schedules: customer sales schedule and optional sales schedule. The customer sales schedule takes into account those engines which are secured by bank guarantees. The optional sales schedule includes the engines which do not have the guarantees yet. Due to the length of the engine production cycle, the customer sales schedule is prepared two years before it is introduced, and the optional sales schedules even three years before the introduction. Obviously, in the meantime the schedules are modified, since owing to the instability of the shipyard industry frequent modifications are necessary.

At the moment, HCP company produces approximately 10 types of engines. An average length of the production process runs at the level of nine months. The manufacturing process in the enterprise takes place in four divisions: welding shop, processing, assembly and packing department.

3 CIMOSA Domain Concept

Enterprise processes of the HCP company were modeled on the basis of the CIMOSA (Computer Integrated Manufacturing Open System Architecture) paradigm. CIMOSA is an open systems architecture for enterprise integration [1]. Its aim is to develop an Open System Architecture for Computer Integrated Manufacturing (CIM) and describe ideas and rules to facilitate the design and construction of future IT systems. This architecture uses the system life cycle concept together with a modeling language and definitions of methodology and supporting technology to cover the function, information, resource and organization aspects.

CIMOSA gives the possibilities to create a model which can be used for process simulation and analysis. It places the business process concept at the heart of the approach to model the various sequences of steps and the numerous flows occurring in enterprises[2]. It is very important, especially for complex production processes. As mentioned before (cf. Section 1), the whole process of marine diesel engine manufacturing is multipronged and composed of a lot of operations.

According to the CIMOSA paradigm, processes can be logically organized into functional clusters which are called domains. They are a modular way to deal with the overall system complexity. Such a domain is a functional scope which embraces complete processes (e.g. management planning domain, procurement domain, loads planning domain, control domain) [2].

Figure 2 shows ten domains (D1, D2, … , D10) which are defined for the engine production process in HCP.

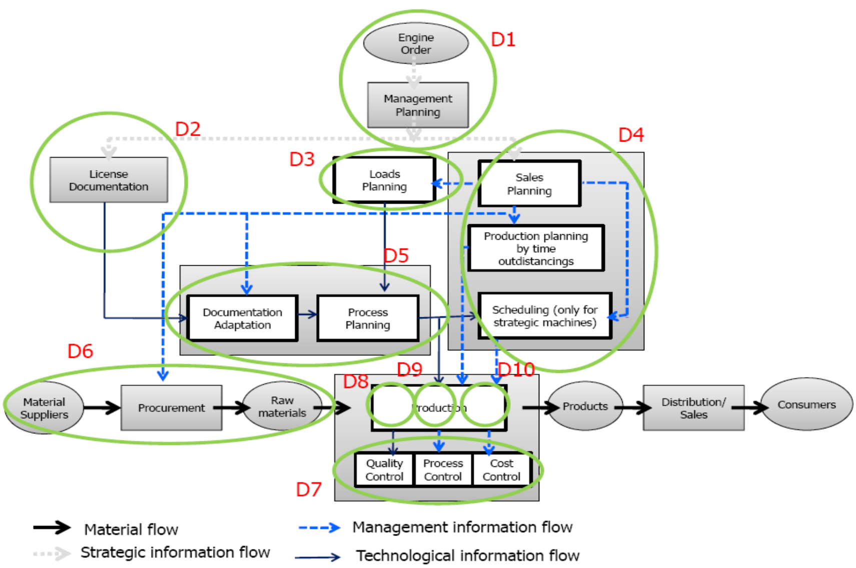


Fig. 2 Domains in marine diesel engine manufacturing process (source: HCP)

4 Multi-agent Approach

Traditional approaches to production planning and scheduling in MRP based logic do not consider real-time machine workloads and shop floor dynamics. Therefore, there is a need for the integration of manufacturing process planning and control systems for generating more realistic and effective plans. The overview of approaches to production planning and scheduling can be found in [10].

Additionally specific, very long cycle production processes are not the main subject which is taken into consideration by this concept. Traditional approaches are to production planning and scheduling are based on: Centralized Optimization Algorithms, Close Loop Optimization and Distributed Process-Planning (DPP) Approaches.

Agent-based approaches provide a distributed intelligent solution by multi-agent negotiation, coordination, and cooperation. The following researches refer to application of multi-agent systems for production planning purpose [10]:

- bidding based approach - the process routes and schedules of a part are accomplished through the contract net bids;
- a multi-agent architecture based on separation of the rough process-planning task as a centralized shop floor planner from the detailed process planning conducted through agent negotiations.
- based on cascading auction protocol provides a framework for integrating process planning and hierarchical shop floor control.

The application of multi-agent can be extended to whole long cycle process due to following potential advantages of distributed manufacturing scheduling [9] logic:

- usage of parallel computation through a large number of processors, which may provide scheduling systems with high efficiency and robustness.
- Ability to integrate manufacturing process planning and scheduling.
- possibility for individual resources to trade off local performance to improve global performance, leading to cooperative scheduling.
- possibility of connection directly to physical devices and execution of real-time dynamic rescheduling with respect to system stability.
- the manufacturing capabilities of manufacturers can be directly connected to each other and optimization is possible at the supply chain level, in addition to the shop floor level and the enterprise level.
- possibility of application of other techniques may be adopted at certain levels for decision-making, for example: simulated annealing, genetic algorithm etc.

Agent-based approaches provide a natural way to integrate planning and control activates and makes possible simultaneously the optimization of these functions. Proposed by authors conceptual framework for the multi-agent approach method involves the hybrid solutions combining the advantages of MRP simple logic and theory of constrains (TOC) ability to synchronize all production and material flow in very long cycle processes. The applications of TOC as synchronization mechanism allows to reduce a number of parameters to be control so it allows to simplify the complexity of integration problem.

5 Multi-agent Approach for Long Cycle Process

Agent-based system is defined in following paper as a multi-agent system that's acts as a support tool and utilized the databases of main system (ERP system).

Multi-agent system is a collection of heterogeneous, encapsulated applications (agents) that participate in the decision making process [7]. The architecture of proposed tool (VLPRO-GRAPH – Very Long Process Graph) is based on the assumption that system will support the MPS creation in ERP system and will be plug in to ERP system database by for example java connector.

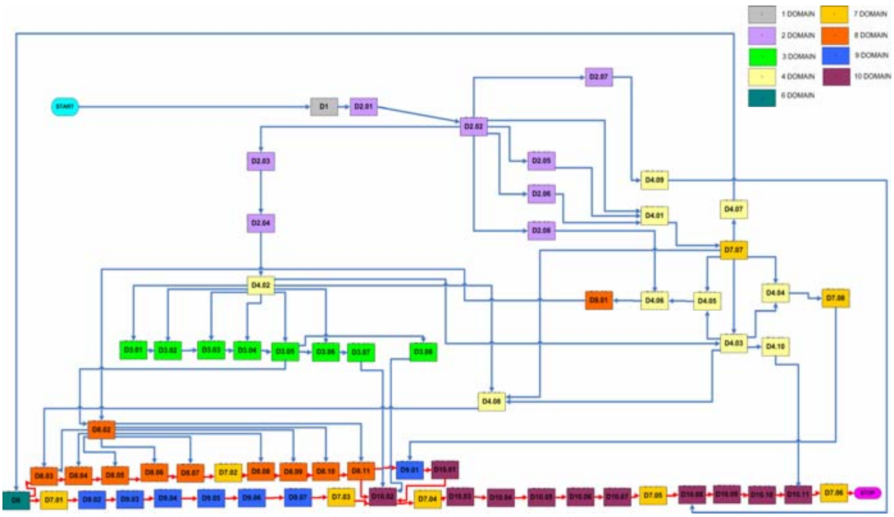


Fig. 3 Model of very long cycle production process of diesel ship engine

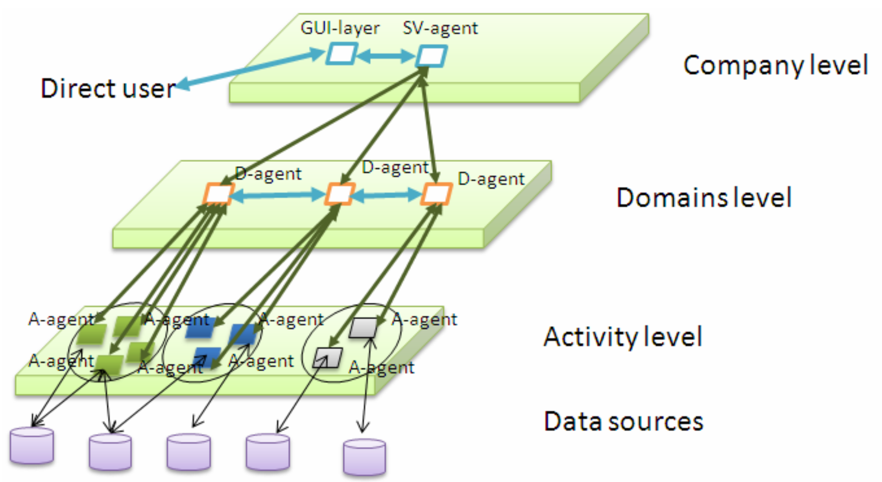


Fig. 4 VLPRO-GRAPH agent model

Figure 3 shows the model and figure 4 illustrates structure and the amount of agents which can be found in each layer. The planning problem in following paper is described at three layers reflecting to [6]: A - Long process perspective so called long process planning; B - The entity level where long process plan is divided to sub-plans which are executed by each subprocess and being transform for individual production schedule at domain level and where local re-planning activities takes place; C - Domain sub-layer where production control activities are executed and information about disturbances are gathered and passed to upper levels.

The graphical user interface agent creates a graphical user interface (GUI) for the interaction of the MAS (Multi-Agent System) to production manager (direct users). The GUI-Agent is able to initialize and sent behavior parameters and messages to the Super-Visor Agent (SV-Agent). The SV-Agent is exactly one in the system because the data from all the domain agents (D-Agent) is fused at this agent to generate re-planning schedules for the production. The SV-Agent is responsible for control of the logic of all agents and creates the plans for the D-Agent. The planning process is presented in figure 5.

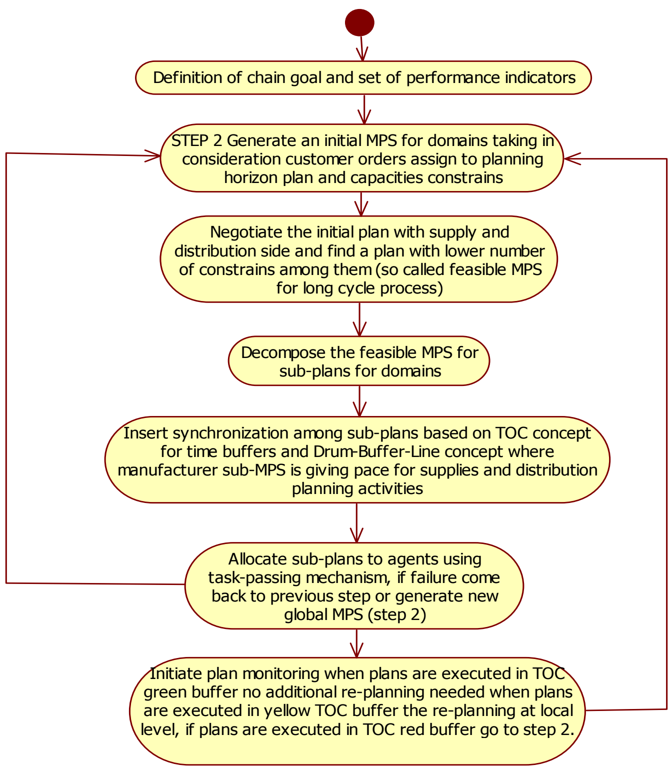


Fig. 5 Planning algorithm for VLPRO-GRAPH agent model

D-Agents are initialized by the SV-Agent and they are responsible for translation of the long process plan into detail schedules. The agent is allowed to prepare the number of alternative (contingency) local plans as long as there are not conflicting with long process MPS. The local re-planning activities are allowed as long as they don't influence the long process MPS. When re-planning activity affects the long process MPS it has to be passed to SV-Agent. The A-Agent is responsible for control of plans execution within sub-process based on given performance indicators. It reports to D-Agent in upper layer whether production plans are executed according to given MPS.

6 Conclusions

In following paper authors present the concept of modeling production proces based on CIMOSA domains idea. The presented concept of the multi agent system on the basis of which a model will be built and used in the simulation experiment will be investigated and described. There are plans to continue this work in collaboration with the ship engines factory where our research began.

References

1. AMICE, CIMOSA – Open System Architecture for CIM, 2nd edn. Springer, Berlin (1993)
2. Berio, G., Vernadat, F.: Enterprise modelling with CIMOSA: functional and organizational aspects. Production Planning & Control 12(2), 128–136 (2001)
3. Chen, D., Doumeingts, G., Vernadat, F.B.: Architectures for enterprise integration and interoperability. Past, present and future. Computers in Industry 59, 647–659 (2008)
4. Jennings, N.R., Wooldridge, M.J.: Applications of Intelligent Agents. In: Agent Technology: Foundations, Applications, and Markets, pp. 3–28. Springer, Heidelberg (1998)
5. Pasek, Z.J., Trujillo, J., Pawlewski, P.: Process Flow Logic Approach to Manufacturing Process Definition, Configuration and Control. In: The proceedings 3rd International Conference on Changeable, Agile, Reconfigurable and Virtual Production, CARV 2009 (2009)
6. Pawlewski, P., Golińska, P., Fertsch, M., Tujillo, J., Pasek, Z.: Multiagent approach for supply chain integration by distributed production planning, scheduling and control system. In: International Symposium on Distributed Computing and Artificial Intelligence 2008 (DCAI 2008). Advances in Soft Computing. Springer, Heidelberg (2008)
7. Pechoucek, M., Říha, A., Vokrínek, J., Marík, V., Prazma, V.: ExPlanTech: Applying Multi-agent Systems in Production Planning. Production Planning and Control 3(3), 116–125 (2003)
8. Saaksvuori, A., Immonen, A.: Product Lifecycle Management, 3rd edn. Springer, Heidelberg (2008)
9. Shen, W.: Distributed manufacturing scheduling using intelligent agent. IEEE Expert /Intell. Syst. 17(1), 88–94 (2002)
10. Shen, W., Wang, L., Hao, Q.: Agent-Based Distributed Manufacturing Process Planning and Scheduling: A state-of-art survey. IEEE Transactions on Systems, Man and Cybernetics-Part C: Applications and Reviews 36(4) (2006)
11. Vernadat, F.B.: Potentials and benefits of Enterprise Architectures: a practitioner's experience. In: Proceedings IESM 2007, Beijing (2007)

A Beehive-Like Approach for Dynamic Generation of Integration Services to Enable Adaptive Supply Chains Using Enterprise Tomography

Ammar Memari, Roberto Pérez López de Castro, and Jan Aalmink

Abstract. Supply Chains have troubles with siwftly adapting to the changing surrounding environment. Because competition is transitioning from among individual companies to among supply networks, visibility of inter-organizational information is critical for rapid response. Information visibility is a key to a successful coordination of events across the network, monitoring analytics that track the health of it and allowing for proactive action. In this paper we propose a multi-agent based approach for dynamic generation of integration services using Enterprise Tomography to index and generate delta trees that can semantically represent accumulative changes in the system and allow for a generative response. Real time availability of information to all members of an SCM System enables them to be agile and in the best position to react quickly, efficiently and collectively to the changing market; permitting them to translate this advantage into better decisions.

1 Fundamentals

1.1 Diversity

Diversity as a general term includes the meaning of variety, but modern interpretations of the word go beyond this meaning to include ethnic, socioeconomic and gender variety. Diversity in this piece of work is thought of as the inclusiveness of variable opinions, beliefs and methods. It is better in many fields, instead of searching for the best and only method of accomplishing a task , to aggregate outputs of all available methods and come out with a collective result. For example, a virus scan in the cloud can be more reliable when several scanners examine the scanned object [19] the same can be said about weather forecasting, credit card verification and other examples found in [21], who claimed that diversity yields more reliable

Ammar Memari · Roberto Pérez López de Castro · Jan Aalmink
University of Oldenburg, Ammerländer Heerstr 114-118 26129 Oldenburg, Germany
e-mail: memari@wi-ol.de, perez@wi-ol.de, ent.tomo@googlemail.com

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 241–248.
springerlink.com

results each time. Well-known examples for utilizing diversity in the domain of information technology are N-Version Programming [8] and Metasearch Engines like dogpile.com, allplus.com and mamma.com.

A main point in the discovery and selection process is the matchmaking where a match or no match is decided between requirements of the request and capabilities of available resources. In many cases a list is returned, ordered by relevance or Degree of Match (DoM). For Example, in the field of Semantic Web Services different mechanisms vary in many respects [22]: Whether to use ranking and DoM, matching parameters, consideration of functional/non-functional properties, service composition support, and matchmaking algorithm type: logic-based or hybrid-based, similarity-matching or graph-based matching. No single matchmaking mechanism has proven to be significantly superior. That implies that aggregating diverse mechanisms and description standards to reach a collective decision is a good practice in this domain.

1.2 Enterprise Tomography

In accordance to tomography in medical diagnostics, we utilize a similar metaphor - the Enterprise Tomography approach. It is supposed to diagnose integration ontologies of a complex SCM (Supply Chain Management) System, perform indexing of all scanned information, and provide time efficient access to the assembled scanned data from different semantic perspectives. Like in medical diagnostics deltas are to be evaluated. In our context integration ontology delta is supposed to be made available for the user community of the Enterprise Software System. Based on this information, the individual teams can evaluate symptoms in a SCM System more easily and they are in a better position to assess the consequences of any

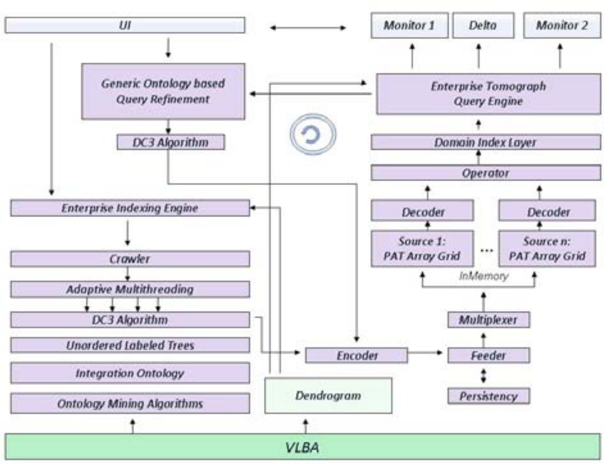


Fig. 1 Anatomy and high level architecture of the Enterprise Tomograph

change in the SCM System software and therefore mitigate the risk of inconsistent data processing.

With a full search UI there is a query to be defined in advance, then the matching search results are listed in a prioritized way. In contrast to this full search approach, the Delta Operator takes only updated, deleted or inserted integration ontologies into consideration. The Delta Operator provides full search capabilities only for those elements. The elements, that are not changed, are excluded by the Delta Operator.

Assuming an end user is performing a business transaction in an Enterprise Information System. What he wants to know is the data footprint on the database created by business transaction execution. With the visualization of this footprint he has basis information on the integration ontologies that are touched by his business transaction. The resulting data footprint on the database is a delta between two snapshots at different points in time. With our delta algorithm we can determine the delta in linear amount of operations, i.e. complexity O(n) for delta index construction [13].

1.3 Semantic Supply Chain Management

Inter-enterprise coordination is a core issue in SCM. The challenge is to allow that every supply network member makes decisions based on the latest and best information from everyone else. However, because of the large number of diverse information systems, the data format (syntax) of each exchange (message) and the meaning (semantics) of both of them usually differs from company to company, or sometimes even within the same company if more than one software product is used as information system [23, 12, 10]. So, users either have to agree on a common model to express the data to be exchanged or they have to individually translate the data received from the business partner to the data format they own. Despite the fact that process modeling languages allow the combination of process definition with Web services orchestration (as process execution structure), they are unable until now to define the integration mechanisms for heterogeneous data schemas. On the other hand, the solutions designed to achieve interoperability of the information systems dont achieve a complete integration since they dont include the integration at processes level.

Business Process Management (BPM) has gained significant attention by both research and industry. However, the degree of mechanization in BPM is still very limited, creating inertia in the necessary evolution and dynamics of business processes. There are several developed and ongoing researches on the topic [9, 14, 2, 5, 3] with the intention to include semantics in the process. On one side there are approaches applying ontologies to describe enterprise models and business processes in general, to show the potential benefits of the application of ontologies for companies; following with the automation of the transition from business process models to monitored execution, and the analysis of what went wrong using the business vocabulary that could be delivered by ontologies; and the latest steps, with attempts to automate processes using SOA and semantic web services.

When facilitating interoperability at the data level one faces the problem that different data models are used as the basis for business formats. For example relational databases are based on the relational model, while XML Schema is basically a hierarchical model [7]. The mapping process is intended to work on a neutral representation, which abstracts from the specific syntax and data model of a particular business schema definition. Therefore, all incoming business schemata are expressed in a neutral format.

2 Problem Definition

Falling margins, globalization, and accelerating innovation cycles are forcing businesses to switch from traditional SCs to adaptive SC networks that possess the flexibility needed to respond to their environment in near real time [1]. For a peer in the chain to be able to respond flexibly, it must be aware of all factors that might affect its decision. We will refer to the set of these factors from here on as the interest domain. This domain is very distributed, the nodes in the network that must be queried to get these interesting items are not only nodes that are known to the user. It is a difficult task to define this domain of interest. A substantial portion of its difficulty comes from its personal nature. interests of a user might not intersect with those of another, that renders the concept of out-of-the-box solutions to be restrictive and archaic since no two users come from the same context.

In order to keep track of the domain of interest, an initial snapshot of the system has to be captured, and then variations should be tracked and responded to. With the advent of the Semantic Web, automated tracking and response can be accomplished by machines that have the ability to understand and process the semantic content. The focus in this paper is on changes in the SC that compromise interoperability. Reasons to change can be found mainly in evolution and infrastructure challenges. Dynamic business conditions spawned by continuously varying demands, competitive pressures and supply arrangements challenge applications to evolve in parallel. A gap between business requirements and configuration of applications can quickly snowball, substantially diminishing application effectiveness. Every upgrade of infrastructure opens questions of compatibility, both backward and forward, requiring business applications to be upgraded.

Keeping track of interoperability issues is a huge undertaking in the complex IT environment of a large enterprise. Conventionally, such problems are addressed and approached in a search engine-like manner. However, in our case of highly dynamic environment where the governance of information is distributed, such an approach might proof inefficient. A search engine would need to index the whole portion of the SC visible to the user peer before it is able to return personalized results; i.e. the personalization step comes too late in the process. It only comes after a valuable amount of time and resources is wasted on crawling and indexing irrelevant information. An integration ontology in this context represents data and software dependencies. To name a few examples for integration ontologies: business object type relations and their depending business object types. A service consumption, i.e.

relations between service consumer and service provider can also be represented as an integration ontology.

3 Approach

Our approach is modularized into three interconnected modules:

3.1 The Beehive Module

We argue that personalization must occur in the crawling phase. i.e. to have a personalized crawling mechanism. For efficiency, crawlers should not crawl and forward to the indexer any information outside the interest domain. Such crawling selectivity requires two abilities in the crawler: ability to distinguish between relevant and irrelevant information, and ability to select the best route leading to relevant information. In our approach, the first is embedded in the crawler by assigning it a semantic goal, and the second is decided collectively by crawlers in a beehive-like manner.

The beehive in our context refers to the pattern of behavior followed by mobile software agents that is analogous to the foraging pattern followed by honeybees, an idea that is based on the works of [17, 18, 16]. The choice to use this non-pheromone-based pattern of foraging came with respect to the research elaborated in [15]. A bee (crawler in this context) is initially dispatched to a peer of the SC, it will run on the remote host to crawl relevant resources (Integration Ontologies). Then it will return to the hive (home peer) and do one of the following: If the quality of the visited host is high enough, the bee will recruite for that host asking other bees to follow it to the pasture containing the advertised host (waggle dance [20]). Otherwise, the bee will enter to the auditorium and watch other bees dancing (recruiting) on the dance floor. Here the bee is faced with a decision to follow one of the dancing bees, or to simply fly to a random new host if none of the advertised hosts is interesting enough.

The different bees in our approach are not clones of each other, they are rather generated initially from diverse prototypes, and optimized using a genetic algorithm. The initiation prototypes hold different matchmaking algorithm to conform to the discussion in section 1.1 , and also differ in other parameters such as their perception of a pasture (neighboring relation-based or Semantic relation-based). Results brought back by generated diverse bees are aggregated in the hive to utilize this diversity in obtaining a reliable decision about the relevance of discovered resources.

After a certain period of working time, performance of each lineage of bees is measured and an evolution iteration step is performed. The system picks the best scoring n lineages to be the parents for the new generation of bees, while other lineages population is reduced gradually. This cycle is user-independent, but the system has also to perform user-specific evolution steps by letting user's feedback influence scores of lineages.

3.2 The Enterprise Tomograph

The pieces of information brought home as bees' payloads constitute a personalized view of the SCM system. This view represents the state of a specific user's interest domain. The view will initially be indexed by the Enterprise Tomograph, and in later steps the Tomograph will take only updated, deleted or added integration ontologies into consideration. First of all the SCM dendrogram is determined. Based on a dendrogram the Enterprise Tomograph starts using the Ontology Mining Algorithm to get a sequence of rooted unordered labeled trees. Basically for a given dendrogram node and a semantic genre a forest of rooted unordered labeled trees is determined. The resulting trees are annotated with tree hash values and with tree node hash values as labels. These trees are used for representing integration concepts, and are sequenced to a textual data, that is indexed with DC3 algorithm.

Each node is annotated with a label, i.e. with a hash value calculated by the ontology mining algorithm mentioned before. Now the question arises which minimal set of operations need to be performed on tree F (original) to retrieve tree G (new). This set is called edit script which serves as a basis for mapping of tree F into tree G in the EnterpriseTomograph Delta Monitor output. Although extremely time consuming procedure for large trees, delta determination for reasonable-sized unordered labeled trees can be performed efficiently on demand.

As a result, the Delta-Operator of Enterprise Tomograph detects within an SCM System created, changed and deleted integration ontologies represented as trees. In addition to detecting such trees, the Delta-Operator calculates the distance between the image of tree before and tree after. Assuming that cardinality of changed trees is small in comparison to the set of unchanged trees, the time consumption of the delta-tree-determination is linear [6].

3.3 The Dynamic Interoperability Framework

After an agent detects a "significant" change and uses the Enterprise Tomograph to get the delta tree, the mapping system is alerted on the fact that a new "translator" is needed. To fulfill this step the mapping system requires three main inputs: the process model, the enterprise ontology and the data schema, which we will further detail. There are several approaches concerning semantic interoperability support for heterogeneous data schemes, among those we decided to follow the patterns set by STASIS [4], a project still in development, but with a high ceiling and expandability.

Once the schema information has been acquired and expressed in a unified model, further analysis and/or processing can be performed to identify a set of mappings between semantic entities being used in different business schemata. The goal is to provide such sets of mappings as input to translator tools to achieve interoperability between dispersed systems without modifying the involved schemata.

The neutral representation of incoming schemata provides the basis for the identification of the relevant semantic entities being the basis of the mapping process; based on the annotation made with respect to the ontologies and on the logic relations identified between these ontologies, reasoning can identify correspondences on the semantic entity level and support the mapping process [7].

The main contribution in this aspect will be the addition of the fourth level of the SCOR process model from each company to the mapping process, based on methodologies described in [11], the hierarchical structure of the SCOR model should prove to be useful to enhance the results of this procedure.

As a result from the mapping system we get a transformation language (could be expressed in XSLT); it will be used to generate the web service which will serve as a translator for the different schemas in the current information interchange. This way seamless interoperation at the process level is achieved including these translators in the executable process model of the workflow.

4 Conclusion

Integration solutions are critical component of todays enterprise strategies, but its a long way between the high-level vision of the adaptive supply network and the basic reality in the development and implementation of these solutions. A contribution to solve this problem is presented by means of a proposal for a framework design that combines the semantic supported modeling with the orchestration of integration processes in the approached context. Agent-based searching is the most suitable approach because it is a convenient way to implement personalized crawling, personalized optimization through evolution, and can cover up for inefficiencies of traditional search and simple subscription.

Enterprise Tomography provides the time and space efficient algorithms and aligned tools to keep track of the domain of interest, taking an initial snapshot of the system, then variations can be tracked and deviations can be calculated using the Delta Operator that can provide delta trees as a form of semantically-described accumulative change scripts.

References

1. Adaptive supply chain network. White paper, SAP AG
2. e3-value, http://www.e3value.com
3. Research project FUSION, http://www.fusionweb.org
4. STASIS project, http://www.stasis-project.net/
5. SUPER integrated project, http://www.ip-super.org/
6. Aalmink, J., Marx Gómez, J.: Enterprise tomography: An efficient approach for Semi-Automatic localization of integration concepts in VLBAs. In: Cruz-Cunha, M.M. (ed.) Social, Managerial, and Organizational Dimensions of Enterprise Information Systems, pp. 232–251. Business Science Reference, Portugal (2009)
7. Beneventano, D., Dahlem, N.: Ontology-driven semantic mapping
8. Chen, L., Avizienis, A.: N-Version Programming: a Fault-Tolerance Approach to Reliability of Software Operation. In: Twenty-Fifth International Symposium on Fault-Tolerant Computing, 1995, Highlights from Twenty-Five Years, p. 113 (1995)

9. Fox, M.S.: The TOVE project towards a Common-Sense model of the enterprise. In: Belli, F., Radermacher, F.J. (eds.) IEA/AIE 1992. LNCS, vol. 604, pp. 25–34. Springer, Heidelberg (1992)
10. Glazner, C.G.: Enterprise integration strategies across virtual extended enterprise networks: a case study of the f-35 joint strike fighter program enterprise. S.M., Massachusetts Institute of Technology (2006); Thesis (S.M.)–Massachusetts Institute of Technology, Engineering Systems Division, Technology and Policy Program (2006)
11. Hahn, A., Haak, L., Abels, S.: Identifying ontology integration methods and their applicability in the context of product classification and knowledge integration tasks. Technical WI-OL-TR-01-2005, Oldenburg (2005)
12. Hohpe, G., Woolf, B.: Enterprise Integration Patterns: Designing, Building, and Deploying Messaging Solutions. Addison-Wesley Professional, Reading (2003)
13. Kärkkäinen, J., Sanders, P., Burkhardt, S.: Linear work suffix array construction. J. ACM 53(6), 918–936 (2006)
14. Lampe, J.C.: Discussion of an ontological analysis of the economic primitives of the extended-REA enterprise information architecture. International Journal of Accounting Information Systems 3(1), 17–34 (2002)
15. Lemmens, N.: To bee or not to bee: a comparative study in swarm intelligence. Master, Maastricht University (2006)
16. Lorenzi, F., dos Santos, D.S., Bazzan, A.L.C.: Negotiation for task allocation among agents in case-base recommender systems: a swarm-intelligence approach. In: Proceedings of the Workshop Multi-Agent Information Retrieval and Recommender Systems (2005)
17. Memari, A., Amer, M., Marx Gómez, J.: A beehive-like multi-agent solution to enhance findability of semantic web services and facilitate personalization within a P2P network. In: ICT Innovations 2009, pp. 227–236. Springer, Heidelberg (2009)
18. Navrat, P., Kovacik, M., Ezzeddine, A.B., Rozinajova, V.: Web search engine working as a bee hive. Web Intelli. and Agent Sys. 6(4), 441–452 (2008)
19. Oberheide, J., Cooke, E., Jahanian, F.: CloudAV: n-version antivirus in the network cloud. In: Proceedings of the 17th conference on Security symposium, pp. 91–106. USENIX Association, San Jose (2008)
20. Seeley, T.D.: The Wisdom of the Hive: The Social Physiology of Honey Bee Colonies. Harvard University Press, Boston (1996)
21. Surowiecki, J.: The Wisdom of Crowds. Anchor (2005)
22. Tsetsos, V., Anagnostopoulos, C., Hadjiefthymiades, S.: Semantic web service discovery: Methods, algorithms, and tools. In: Cardoso, J. (ed.) Semantic Web Services: Theory, Tools and Applications. IGI Global (2007)
23. Weske, M.: Business Process Management: Concepts, Languages, Architectures, 1st edn. Springer, Heidelberg (2007)

Cooperative Purchasing of Logistics Services among Manufacturing Companies Based on Semantic Web and Multi-agent System

Arkadiusz Kawa, Pawel Pawlewski, Paulina Golinska, and Marcin Hajdul

Abstract. The common access to the broadband Internet allows small and medium enterprises (SME) simpler participation in various supply chains [3]. However the main problem still is lack of consistency between processes which are undertaken by various supply chain participants. The aim of this paper is to propose the concept of broader cooperation between manufacturing companies based on semantic web and multi-agent system. The background of this paper refers to previous experiences of the authors by implementation of the CORELOG project. It promotes cooperation in logistics services among manufacturing companies located in the Bologna metropolitan area in order to increase companies' competitiveness as well as environmental sustainability through the rationalization of the logistics processes. The application of semantic web and multi-agent system in the above mentioned project allows the efficient automatic data collection about logistics services providers and their resources, as presented in the Internet.

Keywords: cooperative purchasing, logistics services, e-supply chain, semantic web, multi-agent system.

1 Introduction-Project Background

The project idea starts from the assumption that single company's intelligence does not necessary imply the system's intelligence. In fact, through cooperation companies should rationalize their logistics processes, obtain cost savings and

---

Arkadiusz Kawa
Poznań University of Economics, al. Niepodległości 10, 61-875 Poznan, Poland
e-mail: arkadiusz.kawa@ue.poznan.pl

Pawel Pawlewski · Paulina Golinska
Poznan University of Technology, Strzelecka 11, 60-965 Poznan, Poland
e-mail: {paulina.golinska,pawel.pawlewski}@put.poznan.pl

Marcin Hajdul
Institute of Logistics and Warehousing, Estkowskiego 6, 61-755 Poznan, Poland
e-mail:marcin.hajdul@ilim.poznan.pl

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 249–256.
springerlink.com

reduce empty shipments. At the moment many small and medium sized companies don't activate collaborations as they are traditionally managed like "family enterprises". This limits their ability to get potential opportunities offered by collaboration with other actors operating in the market. Thus, there is a need to create commitment around companies' collaboration through the promotion of public policies, conventions and other knowledge sharing moments.

In described case, manufacturing companies decided to start with a common purchasing approach that stands out for less organizational costs and high savings due to the growing companies' bargaining power and the rationalization of logistics providers. Through the pilot implementation the involved manufacturing and transport companies achieved significant cost savings. These excellent results are actually the starting point for the involvement of further companies and the implementation of other collaborative solutions among SMEs in the Emilia-Romagna Region.

The pilot implementations which were carried out with the CORELOG project (financed under the Interreg IIIB CADSES EU Initiative), included I-LOG (Industrial Logistics and intermodal transport for SMEs' development) results with the objective to implement, in a regional scale, the already developed medium and long term strategy.

In collaboration with local associations, 5 companies, working in the mechanical district, were involved on a voluntary basis.

The main activities realized were:

- review of companies' strategies for the purchasing of transport and logistics services;
- analysis of companies' inbound and outbound flows and related costs;
- verification of the scope of the coordinated purchasing (inbound and/or outbound, or by traffic axis) and estimation of the possible savings;
- development of an action plan for the realization of the proposed strategies and meetings with logistics providers to discuss its feasibility;
- definition of the medium and long term actions for evolving from common purchasing to integrated planning.

As a result of the carried out activities companies obtained savings from 53% to 61% of costs through the common purchasing of transport services in the Bologna-Milano route.

The project was successful but there is still a big area within it with improvement potential. There are still many problems in the data collection phase. In fact, the companies involved in the pilot project haven't got the necessary information in their IT systems. This problem is common to the majority of SME's, as in many cases they don't perform regular reporting and analysis of logistics flows. SME's haven't integrated information systems that allow monitoring of data related to the inbound freight flows. Moreover, as inbound flows are linked to the company's production system, it is more difficult to implement collaborative solutions, as this would require significant changes in the planning and production process.

Another problem regarding project's results implementation is the fact that the initial elaborated solution is dedicated to companies, which are located in the proximity. There is no possibility to start cooperation with companies located in a bigger distance due to the lack of automated searching, selection and settling the initial cooperation conditions with partners.

In order to solve the above mentioned problems a new model called MASEW (Multi-Agent based on Semantic Web) is proposed.

2 MASEW Model

The MASEW model is a hybrid solution elaborated based on agent technology and semantic web concept including such elements, as XML, Resource description Framework (RDF), and ontologies. Semantic web is an initiative that aims to introduce standard description of the content in Internet in order to allow computer programs (eg. agents) processing of data and information appropriate to their purpose and meaning [1].

The main feature of semantic web is its interoperability. It is understood as an ability of different information systems to cooperate, safely exchange data with predefined structure, as well as the further mutual usage of this data in order to create information. Communication between IT systems is possible independently of programming languages, platforms and operations systems they use, and applied information exchange standard. All the above mentioned features make the concept of semantic web especially suitable for application in the e-supply chains.

Types of Agents in MASEW Model

In the distributed environment e.g. Internet two types of managing/leader agents are common, namely stationary agent (that has in command the other agents) and PSA (Polymorphic Self-Slimming Agent) with bimodal structure (an agent is composed of a sort of agent head, called *bootstrap agent*, and agent body, called *proxy agent*) [4]. In the MASEW model the first type of managing agent is implemented. In order to gain information the Leader Agent (LA) delegates collecting of predefined information scope about potential business partners to group of mobile agents. LA represents the particular interests of cooperating companies and it is responsible for cooperation, communication, negotiation with the other agents representing companies outside the cooperation network. On the basis of collected information the LA makes decisions regarding follow-up actions. Leader Agent is able to manage complex business processes. It cooperates with groups of agents and creates multi-agent system, as followed:

- Searching Agent (SA) – it is one the most mobile agents, which is responsible for searching in the Internet suitable business partners and trade partners according to the criteria defined in cooperation scenario. It is able to inform the Verifying Agent (see below) about semantic inconsistency of found information.

- Offering Agent (OA) – represents companies (e.g. n-trier suppliers, logistics companies, distribution centers), which offer particular products and services
- Informing Agent (IA) – it is responsible for sending information to LA about new attractive offers available (e.g. in e-market server or www), as well as for informing agents representing trade partners about the possibility of starting cooperation with LA after fulfillment of defined criteria.
- Negotiating Agent (NA) – represents trade partners and on behalf of them it negotiates cooperation's conditions with LA.
- Verifying Agent (VA) – it verifies ontologies and acts together with tool for semantic verification of information placed in the web by trading partners.

Agents representing particular companies cooperate together and negotiate cooperation conditions in order to reach common goal and at the same time to fulfill the outlines of delegating institutions. It should be mentioned that agents representing different, sometimes competing companies can build up temporary alliances in order to reduce the cost of logistics services (e.g. combined transportation). At the same time the logistics services providers could cooperate in order to fulfill the requirements stated in the offer enquiry by client (e.g. when single company does not have enough capacity). The communication between agents and user is asynchronic, it means that the confirmation of message by receiver is not required in order to continue the process execution. Agents and users are informed automatically about any new event e.g. new offer arrival.

The access to the semantic web does not require application of any specialized IT systems. Information can be process in format that is readable and understandable for computer and any user. This solution gives SMEs a better possibility to capture information and exchange them within particular enterprises' network, as well as it helps to build up business relations.

Searching for potential business partners by agents in semantic web, can be done superficial or in-depth. In the first case agents search only on these web sites with words with strict defined meaning, not key words which are often ambiguous. In the second case agent searches more precisely, analyses particular web sites content and passes the results to other agents. The first solution is quicker but definitely less precise than the second one.

MASEW Model Implementation – Example

In this example it is assumed that three companies (A, B, C) seek for carrier, which will organize freight from Bologna to Milano in Italy (see Fig.1.). These companies have particular requirements regarding: mean of transportation (truck with loading capacity of 24 t), delivery cycle 1 day and delayed payment period up to 60 days. The pickup places are manufacturing sites at three dispersed locations in Bologna, the places of delivery are three distribution centers in Milano. The additional requirement is the participation of the carrier in express delivery forum.

The organization of such process by hand would be time consuming due to the fact that a company would have to visit web pages of all potential carriers, as well as web site of the express delivery providers' forum in order to verify whether particular carrier is associated there. In the next step all potential carriers which don't have appropriate mean of transportation should be excluded. Additionally companies A, B, C would need to agree before all conditions are settled and approved all the changes between present and initial adjustments.

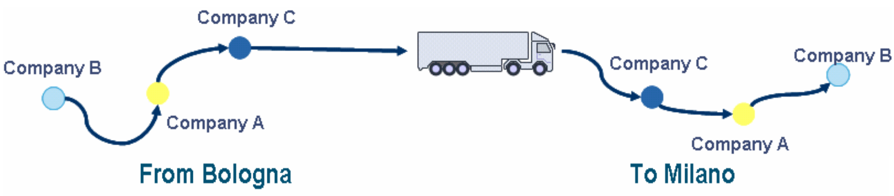


Fig. 1 Companies' cooperation by transport operations

Application of the multi-agent system and semantic web facilitates the above mentioned process. The procedure below presents the process of finding business partner by agents, following the defined scenario (see Fig. 2):

1. Companies (A, B, C) (or their IT systems) order their searching agents (SA) execution of task (in this case it is defined as a choice of carrier for shipment of goods from Bologna to Milano according to optimization criteria).
2. SA through hyperlinks finds ontology defining key words. Then it communicates with OA representing potential suppliers, carriers, etc. and collects their offers and passes them to LA for analysis.
3. In case when SA has doubts about the content of the offer, it could command Verifying Agent (VA) to verify ontology and to contact its issuer.
4. Before offer choice LA questions supplier agent OA whether the offer is still valid. If not, then the whole procedure needs to be repeated. If the offer is still valid, then the appropriate resources are booked. LA can book multiple carriers, especially in case when individual carrier is not able to fulfill demand. Particular OA (especially competitors) can create cooperation network (clustering).
5. After the offer is chosen LA informs agents representing business partners (NA) and starts negotiations regarding number of routes, delivery dates, prices etc.
6. If the negotiations are successful, then the cooperation starts and agents sign the commission of service purchase on behalf of companies they represent.
7. When the cooperation is finished, LA stores the scenario in its database, such practice helps companies (A, B, C) to used successful scenario in the future.

All parties involved in cooperation are informed about every pre-transactional action and any identified disturbances in the process of order fulfillment execution (e.g. delays, lack of delivery, damages and returns).

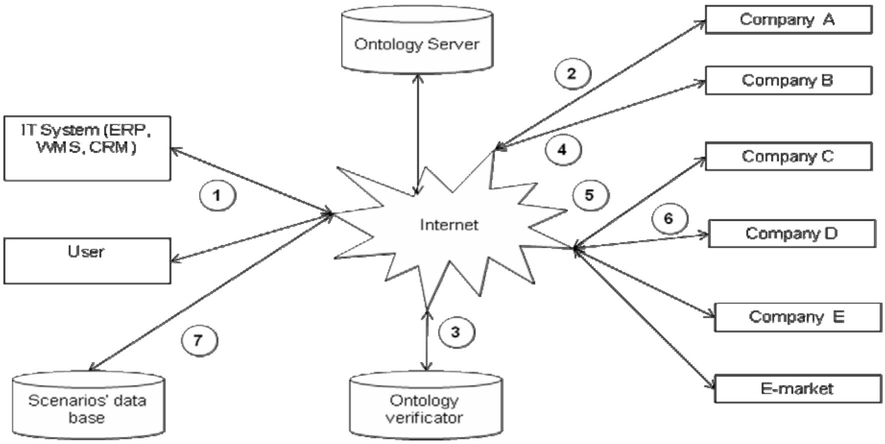


Fig. 2 Model of multi-agent system based on the semantic web idea

Resources Description in MASEW Model

The key factor in multi-agents systems is the homogeneous communication. It secures efficient common goal fulfillment and reduces misunderstandings. In order to execute the above mentioned algorithm the communication between agents representing different companies is needed. In the MASEW model agents exchange messages based on ACL (Agent Communications Language) using FIPA protocol, especially from FIPA Communicative Act Library Specification [2], which provides cooperation between agents systems. These protocols define syntactic, semantic, and pragmatic of created messages. This solution helps to send messages between independently designed and developed agent systems. The structure of ACL with RDF message is presented below. It is an answer to query „Query Ref" number *query00001*, regarding the choice of carrier for delivery of goods from Bologna to Milano in Italy.

The message type „Inform" is defined in standard "FIPA Communicative Act Library Specification" [2]. The message is based on „logistics-services-providers" ontology, including the specification of logistics terms and their meanings. The sender to this message is SA, which passes the answer to LA's query, with information about finding the appropriate logistics services provider placed in Bologna.

Example of the message „Inform" type in the FIPA protocol with RDF.

```
(inform
  :sender (agent-identifier :name SA)
  :receiver (set (agent-identifier :name LA))
```

```
    :content
      "(result
        (action (find-resource : query...))
        ("<?xml version="1.0"?>
      <rdf:RDF>
       <res:Resource rdf:ID="BestDeliveryLtd">
        <loc:city>Bologne
        </loc:city>
        <loc:country>Italy
        </loc:country>
        <loc:streetAddress>14, Belle
        </loc:streetAddress>
        <loc:phone>+33 61 11 22 555
        </loc:phone>
       </res:Resource>
       ...
  </rdf:RDF>")
    :language fipa-sl
    :ontology logistics-services-providers
    :in-reply-to query00001)
```

In MASEW model Resource Description Framework (RDF) and ontologies are used at any stage of tasks execution of finding business partner by agents. It speeds up the searching process because all data is well structured. As a result easer data filtering and elimination of unnecessary data is possible. Resource description is presented below. This example includes the description of found mean of transportation in XML format including meaning of this description in RDF. Schema *vCard* was applied for resource description. The resource identification number „LSP ID: H17N13M2T" is unique parameter, as well as the e-mail address "Roberto Luci".

Example of the resource description in XML with their meaning in RDF.

```
  <?xml version="1.0"?>
  <rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-
       ns#"
  xmlns:dc=http://purl.org/dc/elements/1.1/"
  xmlns:vCard = "http://www.w3.org/2001/vcard-rdf/3.0#">
    <rdf:Description rdf:about="LSP ID: H17N13M2T">
      <dc:title>Truck 24 t</dc:title>
      <dc:creator rdf:resource="mailto:luci@bestde.it">
      <dc:subject>Transport, Bologne</dc:subject>
      <dc:description>Truck Box</dc:description>
      <dc:publisher>Best Delivery Ltd.</dc:publisher>
      <dc:contributor>Company A</dc:contributor>
      <dc:date>2009-11-15</dc:date>
      <dc:identifier>H17N13M2T</dc:identifier>
      <dc:language>Eng.</dc:language>
      <dc:coverage>Italy</dc:coverage>
    </rdf:Description>
    <rdf:Description rdf:about="mailto:luci@bestde.it">
  <vCard:FN>Roberto Luci</vCard:FN>
```

```
<vCard:Title>IT specialist</vCard:Title>
<vCard:Role>Programmer</vCard:ROLE>
<vCard:ORG>
        <vCard:Orgname>Best Delivery</vCard:Orgname>
        <vCard:Orgunit>IT Department</vCard:Orgunit>
</vCard:ORG>
<vCard:TEL rdf:parseType="Resource">
        <rdf:value>+33 61 11 22 555</rdf:value>
</vCard:TEL>
<vCard:ADR rdf:parseType="Resource">
        <vCard:Street>14, Belle</vCard:Street>
        <vCard:Locality>Bologne</vCard:Locality>
        <vCard:Pcode>00-111</vCard:Pcode>
        <vCard:Country>Italy</vCard:Country>
      </vCard:ADR>
  </rdf:Description>
</rdf:RDF>
```

3 Conclusions

Almost unlimited access to the Internet makes possible cooperation in the area of logistics process not only between big company but also SMEs. The lack of consistency of business process performed by particular entities and the variety IT systems used by companies, cause problems with automatic partners networking. The presented by authors MASEW model is solution that enables efficient creation and automatic collection of data about companies and their resources published in the Internet. It broadens the possibilities of initial CORELOG project's results implementation. Agents representing particular companies coordinate and establish cooperation's conditions, in order to reach common goal of cost effectiveness.

References

1. Berners-Lee, T., Hendler, J., Lassila, O.: A New Form of Web Content that is Meaningful to Computers will Unleash a Revolution of New Possibilities. Scientific American 284(5) (2001)
2. FIPA Communicative Act Library Specification, http://www.fipa.org/specs/fipa00037
3. Fuks, K., Kawa, A., Wieczerzycki, W.: Dynamic Configuration and Management of e-Supply Chains Based on Internet Public Registries Visited by Clusters of Software Agents. In: Mařík, V., Vyatkin, V., Colombo, A.W. (eds.) HoloMAS 2007. LNCS (LNAI), vol. 4659, pp. 281–292. Springer, Heidelberg (2007)
4. Wieczerzycki, W.: Polymorphic Agent Clusters – the Concept to Design Multi-Agent Environments Supporting Business Activities. In: Mařík, V., William Brennan, R., Pěchouček, M. (eds.) HoloMAS 2005. LNCS (LNAI), vol. 3593, pp. 233–245. Springer, Heidelberg (2005)

Sandbox for Development of Evolving Manufacturing Control Architectures

Zbigniew J. Pasek

Abstract. Control architectures for manufacturing cover a broad range of categories from hierarchical at one extreme to purely heterarchical on the other spanning a broad variety of solutions, from quasi-heterarchical structures to self-organizing systems. This paper describes an agent-based framework that provides the tools and components for fast implementation and testing of various control structures in a manufacturing environment. Typical components include, for example, a generic agent for encapsulation of machine-level control, a configurable decision process for the agents, hierarchical or heterarchical control frameworks. Use of these predefined components enables rapid prototyping and instantiation of system-level control.

Keywords: Manufacturing, Control, Agent-based, Reconfiguration.

1 Introduction

To stay competitive, manufacturing companies must develop and manage assets enabling production of goods with top productivity and minimal cost. At the same time these systems must exhibit rapid responsiveness to the frequent and unpredictable market variations. A new paradigm of Reconfigurable Manufacturing Systems (RMS) [4] facilitates most of these challenges by quick adjustment of production capacity and functionality, enabled by rapid structural system changes, both at the hardware and software levels. Achieving RMS goals requires fast and effective integration of software and hardware components, not only at the manufacturing floor level but also with the rest of the enterprise. This paper presents an implementation, development and testing framework that provides the components to rapidly implement any control structure suitable for a manufacturing environment.

Zbigniew J. Pasek
Industrial and Manufacturing Systems Engineering Dept.
University of Windsor
401 Sunset Ave., N9B3P4 Windsor, ON, Canada
e-mail: zjpasek@uwindsor.ca

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 257–265.
springerlink.com

2 Background

Varying market demands, environmental/societal requirements, and continuing progress of technology define a need for manufacturing control systems that are capable of managing changes effectively and efficiently [11]. Such control system architectures must be interoperable and capable of handling a wide variety of data models and applications, as well as conform to ever increasing demands on performance requirements of process control applications [7]. The hierarchical and heterarchical systems define two ends of the spectrum for control system organization [3, 10]. Hierarchical systems are known to deliver predictable performance, but being based on a set of fixed rules offer little in terms of flexible strategies. The heterarchical systems, on the other hand, inspired by biological analogies and market economies, offer improved responsiveness to disturbances, but often at the cost of sub-optimal solutions and lack of predictability. Another class of approaches [1, 2], holonic control systems, attempts to combine advantages of hierarchy in distributed structure. Only recently the commercial control systems based on multi-agent technologies have become available [5], but their practical acceptance is still fairly limited. Nevertheless, despite these developments being still in early stages, they offer an exciting vision of self-organizing and self-modifying systems capable of highly responsive behavior.

3 Framework

The proposed framework consists of a generalized agent class that can be programmed to interact with both the heterarchical and hierarchical control structures. Hierarchical systems consist of tree-structures of control nodes that deterministically govern the actions of the leaf nodes (machines or services). Heterarchical structures provide a mechanism for parts and machines to reach agreements on operations. For example, a part could "buy" services from a machine that offers the lowest cost.

The goal of the framework is to provide rapid prototyping and implementation of any hybrid shop-floor control structure. To accomplish this task, some key aspects of the system have to be defined. These include: the representation of the machines/parts, the heterarchical control system, the hierarchical control system, and the way in which the agents interacted within each of these systems.

3.1 Agent Representation

The goal in both a hierarchical control structure and a heterarchical control system is to provide a software abstraction of the machine/part so that it can be easily added to, or removed from, the control structure. This abstraction can be defined most precisely as an encapsulation of the machine-level control behind a standard interface [6]. Such interface allows the modules in the system to act as plug-and-play entities.

The agents communicate and act within the control structure without exposing the details of their operations. This information segregation prevents interdependencies between the agents and also adds to the robustness of the system. The agent framework (Fig. 1) used for both machine and part agents, has three main elements:

- **Machine Controller Interface:** The agent construct has an interface to the physical machine to provide control of the machine, which interacts with the API provided by the machine manufacturer. The machine level control was encapsulated behind a simple start, stop, resume, and reset interface [9]. Due to the proprietary nature of the machine controller interface, the framework agent simply provides skeleton code for the simple operations start, stop, resume, and reset. Calls to the proprietary library must be inserted into these functions. Part (product) agents are basic software instantiations of the part in the system, and they do not make any use of the machine interface.
- **Master-Slave Control:** Similar to how the machine interface is responsible for translating simple commands at the machine level, the Master-Slave Control is responsible for handling those commands at the agent level. The start, stop, resume, reset functionality is self-explanatory and the simplest implementation forwards these commands to the machine interface. In addition to forwarding calls, other user-defined, agent-related processes should be controlled within these functions. These include, for example, data monitoring, coordination, diagnostics and maintenance.
- **Bidding Control:** The bidding control module is responsible for interacting with the heterarchical control framework and making calls into the Master-Slave Control Module. The Bidding Control module defines actions related to the bidding processes. The bidding functionality and server callback routines are used to interact with auctions on the server. The agent's strategy is defined by how it reacts to events within the auctions it is competing in. Part agents use the auction creation routines to set up auctions for the operations needed for their processing. Machine agent may also be programmed to use these routines, to build auctions for maintenance requests and other services.

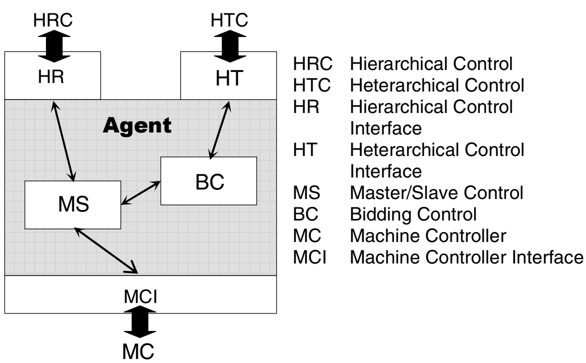


Fig. 1 Agent Framework

The hierarchical framework interface communicates using CORBA (a communication middleware abstraction, located between the application and network layers, which rests upon TCP/IP). The heterarchical framework interface uses XML to communicate with the heterarchical server. Both interfaces provide the corresponding remote procedure calls of the functions mentioned in the Master-Slave Control Module and the Bidding Control Module.

3.2 Heterarchical Control Infrastructure

Many, recently developed, heterarchical systems (Fig. 2) rely upon an auction strategy similar to Contract-Net [8]. The heterarchical control used by this framework was built mostly in Java. It uses XML to communicate with the agents within the system through their corresponding interfaces. Thus all the Bidding Control functionality runs through this infrastructure. On the server side, the system works very similar to AuctionBot [12] to setup and run auctions.

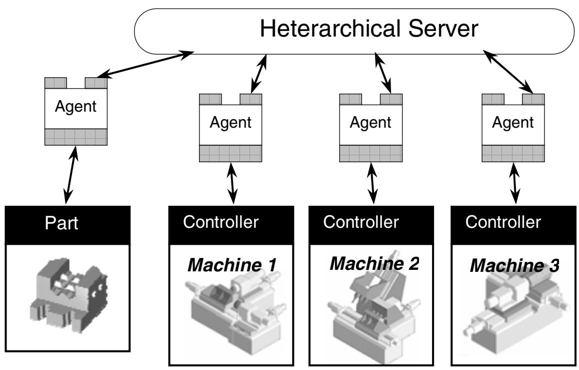


Fig. 2 Heterarchical Control Architecture

A simple implementation of purely heterarchical control in this framework is carried out as follows:

1. The part agent sets up auctions for its desired operation(s) using the *createAuction* routine.
2. The machine agents poll the heterarchical server for new auctions corresponding to operations they can perform using the *getAuctions* routine.
3. Machine agents bid on the corresponding returned auctions using the *commitBid* routine.
4. When outbid, machine agents are notified through the *outbid* routine. A machine agent may choose to resubmit bids based upon the *getAuctionPrices* routine.
5. The part agent determines when to close the auction based upon the prices it sees through the *getAuctionPrices* routine. The part agents then calls the

endAuction routine which returns an allocation containing which part was allocated to which machine.

6. Upon closing the auction an *auctionWon* or *auctionEnded* message is sent to the machines that bid in the auction.

The user interfaces are additional components of the heterarchical system framework: Figure 3 displays the collection of machine agents that are online and interacting with the server. A user can take machine agents on and offline and observe the bidding history and number of parts in an agents queue using the interface. Corresponding server interface (not shown) contains information about all of the auctions occurring in the system, tracks machine agents participating in the auctions and the parts being auctioned off.

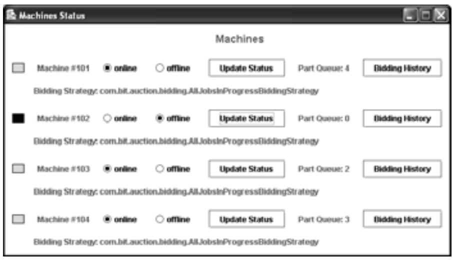


Fig. 3 Machine Status Interface

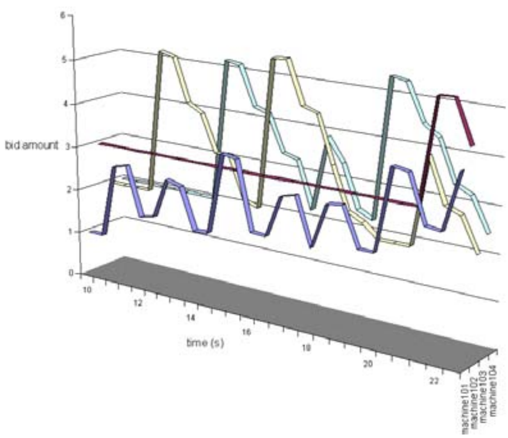


Fig. 4 Bid amounts over time

Figure 4 shows how bidding values of machine agents vary over time. The scenario consisted of submitting 26 part agents requiring the same operation to the server. The parts are auctioned off two at a time. Due to the bidding strategy employed by the machine agents, as they win auctions, they bid higher because of the aggregate cost to machine the part(s) it just won and the additional cost of the operation its bidding for. The bid amount drops as the machine progresses through the operation allocated to it.

3.3 Hierarchical Control Infrastructure

The hierarchical control infrastructure, built using Java and CORBA, allows for control structures such as those pictured in Figure 5.

A simple implementation of a purely hierarchical architecture consists of tree structures built out of software control agents. These distributed internal nodes run simple deterministic control, which dictates when to activate child nodes. Internal control nodes can be built from the same basic framework as the machine agents. In the Master-Slave Control Module's basic functions the coordinating nodes can be programmed to make function calls through the hierarchical interface to start their children nodes instead of communicating to a machine. The leaf nodes of the hierarchy are the true machine agents which actually start physical machines.

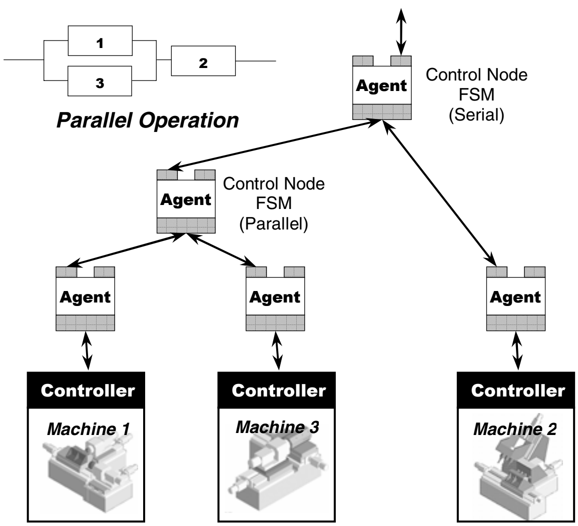


Fig. 5 Hierarchical Architecture

3.4 Comparing Control Structures

The introduced framework provides a "toolbox" for building different control structures. Assessing their suitability is yet another matter. In manufacturing, most common evaluation metrics pertain to part flow and machining time [9]. The part flow metric combines time delays in the control system with time delays in the part movement. To decouple these metrics, the heterarchical server logs separately allocations and auction durations. Using such data, the effects of the heterarchical aspects of a system in terms of time delay and allocation efficiency can be measured.

Another issue concerns flexibility and autonomy versus optimality. The most appropriate control structure depends highly upon the features of the system it is controlling. For example, if the resource pool is ever changing (due to machine failures, part type changes, capacity fluctuation), a more flexible heterarchical-like control structure is desired. If the process is well defined, consistent, and optimality is the major concern then a hierarchical-like structure is a better fit.

4 Example Structure

An example of a self-organizing system, blending the advantages of hierarchical and heterarchical architectures is described, built using the proposed framework, is described. A system is self-organizing if the structure of the system grows from the system itself [1]. The following example shows a way to "grow" the control structure based upon the local interaction of the pieces rather than a global observer specifying its organization. The self-organizing structure starts as a purely heterarchical structure. The evolution that follows is based on recognition by machine agents commonality of patterns in operation sequences. Individual bidding for each operation is replaced with a deterministic sub-hierarchy that has one agent bidding for the sequence of operations. Such groupings reduces both the number of auctions and the number of participating agents, and reduces the bidding-related time overall delay.

The control structure uses three basic agent types: for parts, machines, and coordination. Part agent interfaced to the heterarchical control system sets up auctions for its desired operations and records all the allocations that it receives.

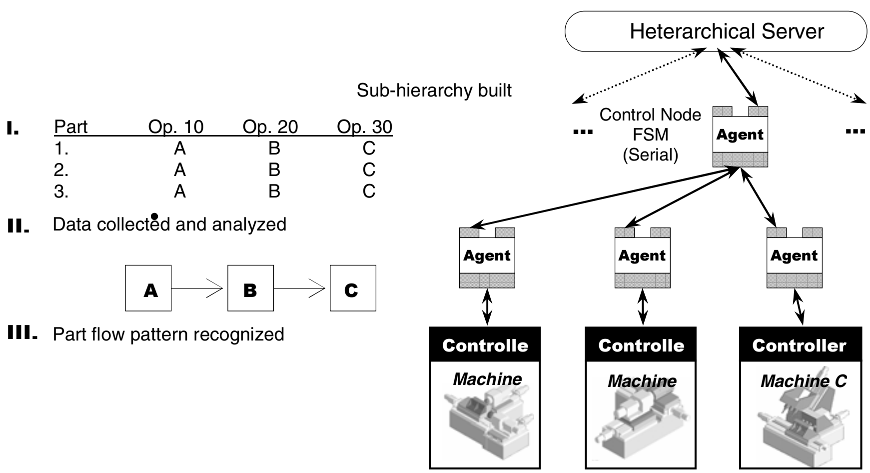


Fig. 6 Example of sub-hierarchy construction

Machine agent is a simple heterarchical agent performing basic bidding operations; it also communicates with all part agents for auctions it wins and requests their allocation history. Given this history, the agent builds aggregate hierarchies if appropriate.

Figure 6 shows an example where the machine agent creates a serial line configuration. By examining the allocations of parts that it is winning, Agent C recognizes sequential parts flow. It then launches a serial coordinator as a separate process. Agent C gets the reference to this new serial coordinator (*getReference*) and also grabs the references to Agents A and B. It makes remote calls through the heterarchical interface to the serial coordinator to add Agents A, B, and C as children. Then Agent C makes *changeControlMode* calls to A, B and C to move them (and itself) into a purely deterministic operation mode. Now a single serial controller, representing a group of three machines, is able to make a bid.

5 Summary and Future Work

This paper outlines concept of a framework for fast prototyping of manufacturing control structures. The framework is centered around a generalized, programmable agent construct, which interacts with the provided hierarchical and heterarchical control structures according to how it is programmed. Thus any control structure from the hierarchical to heterarchical spectrum of can be built. Initially, the serial coordinator was pre-built with the serial recognition ability hard-coded into the machine agent's *auctionWon* routine. To abstract out this functionality from the coordinators an inference engine, able to recognize a pattern in part diagram, is being built, which in turn, can be used to automatically generate source code to carry out the actual pattern matching.

References

1. Bartholdi, J., Eisenstein, D.: A Production Line that Balances Itself. Operations Research 44(1), 21–34 (1996)
2. Bongaerts, L., et al.: Hierarchy in distributed shop floor control. Computers in Industry 43, 123–137 (2000)
3. Dilts, D., et al.: The Evolution of Control Architectures for Automated Manufacturing Systems. Journal of Manufacturing Systems 10(1), 79–93 (1991)
4. Koren, Y., et al.: Reconfigurable Manufacturing Systems. Annals of the CIRP 48(2), 527–540 (1999)
5. Mařik, V., et al.: Rockwell Automation Agents for Manufacturing. In: 4th Intl. Conference on Autonomous Agents, pp. 107–115 (2005)
6. Pasek, Z.J., et al.: Infrastructure for System-Level Configuration and Control of Manufacturing Devices. In: Japan-USA Symp. on Flexible Automation, vol. 2, pp. 1389–1392 (2000)
7. Pritschow, G., et al.: Open Controller Architecture – Past, Present and Future. Annals of the CIRP 50(2), 463–470 (2001)
8. Smith, R.G.: The Contract Net Protocol: High-level Communication and Control in a Distributed Problem Solver. IEEE Trans. on Comp. 29(12), 1104–1113 (1980)

9. Vaario, J., Ueda, K.: Self-Organizing in Manufacturing Systems. In: Japan/USA Symposium on Flexible Automation, vol. 2, pp. 1481–1484 (1996)
10. Van Brussel, H., et al.: Reference Architecture for Holonic Manufacturing Systems. Computers in Industry 37(3), 255–274 (1998)
11. Wiendahl, H.-P., Scholtissek, P.: Management and Control of Complexity in Manufacturing. CIRP Ann. 43(2), 533–540 (1994)
12. Wurman, P., Wellman, M., Walsh, W.: The Michigan Internet AuctionBot: A configurable auction server for human and software agents. In: Proc. 2nd Intl. Conf. on Autonomous Agents, pp. 301–308 (1998)

Agent-Based Model of Kanban Flows in the Environment with High Demand Variances

Paulina Golińska, Joanna Oleśków-Szłapka, Agnieszka Stachowiak, and Paweł Rudiak

Abstract. The following paper introduces a very interesting and quite common problem of dealing with high demand variance. The company, in which the problem was identified, is of automotive industry and it produces elements of vehicles interior furnishing. The flows of materials and information in the company are presented and discussed and kanbans used to manage the flows are introduced. The solution of the problem is presented, as well as its model based on multi-agent model. The model is to be used in simulation testing efficiency of solution suggested in dynamic environment.

Keywords: Kanban, High-demand, Agent-based model.

1 Introduction

Manufacturing enterprises are facing a substantially more complex situation than ever before because of the unpredictable market demands, growing product customizations, and fluctuating production environments. In order to stay competitive in the market, the decision making processes need to have the ability to adapt to the changing environment and to handle the system complexity. The ability to make use of the system flexibility and to retain flexibility itself is a primary requirement for the decision tools to be effective [1].

From our observation, even in a well designed assembly line, when the whole line is optimized, including its layout, processes, batching, scheduling, and operations, material handling is still laid outside of the scope of control. In certain industrial

Paulina Golińska · Joanna Oleśków-Szłapka · Agnieszka Stachowiak · Paweł Rudiak
Institute of Management Engineering,
Department of Informatics and Management,
Poznan University of Technology
Strzelecka 11, 60-965 Poznań, Poland
e-mail: paulina.golinska@put.poznan.pl,
joanna.oleskow-szlapka@put.poznan.pl, a_stachowiak@wp.pl,
pawelrudiak@plusnet.pl

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 267–275.
springerlink.com

sectors, material handling has been the major barrier that results in production breakdowns, low efficiency, and low performance of a production system.

Intelligent agent technology has recently been applied to material handling system simulations [3]. Since its emergence, agent technology has been widely recognized as a promising paradigm for the next generation of design and manufacturing systems [3]. Agent technology makes a perfect fit for modeling dynamic and adaptive manufacturing systems. There are two types of production control systems, namely push and pull. MRP systems and Kanban control systems are the two most popular implementations of the push and pull strategies, respectively. Pull is essentially a replenishment strategy that was initially designed for manufacturing environments producing repetitive products with high volumes [7].

The majority of papers related to Kanban problems treat the analysis problem. Papers dealing with both performance analysis problems and the derivation of structural properties belong to this group. For a given Kanban system they investigate important steady-state performance measures such as throughput, average Work-In-Process (WIP), and average flow time of items. The used mathematical models include as well analytical models as simulation models.

However, the Kanban in its reactive form cannot cope with large sudden changes in demand, greater than 10%, say [5]. This is because the number of Kanban required in circulation for good operation of the system depends quite critically upon the item demand and supply capacity. The standard solution advocated under the Toyota Production System [6] is to manipulate the schedule of end-items being assembled so that there is a nearly constant mix of items required each day. However, in the Electronic Commerce era, with increasing emphasis being placed on responsiveness to consumer needs and particularly mass-customization of products, this demand smoothing approach to protecting the Kanban systems from fluctuations may not be desirable. For this reason, some automotive manufacturers are seeking to add "intelligence" to the otherwise reactive Kanban system.

Therefore a technology that can be used in material handling system is an agent. In JIT material handling simulation system, multiple agents can be implemented to facilitate a collaborative problem solving environment. For example, each transportation vehicle is encapsulated as an agent so that it is manageable on its own parameters and behaviors, such as velocity, local schedule, and the associated scheduling, routing and conflict resolving rules [4].

Various simulations were done on Kanban-related topics targeting different purposes [10]. For example:

1. To study Kanban allocation problems, either for a fixed allocation of Kanbans [11] or a flexible Kanban allocation strategy [2,9].
2. To look at Kanban control issues in a Kanban-based system, such as Kanban sequencing [12,13], buffer capacity [14], control, and maintenance policies [15].
3. To conduct performance analysis studies of Kanban-based systems [16,17].
4. To compare the Kanban control mechanism with others, i.e. traditional push systems [7].
5. To combine the simulation with optimization approaches, such as analytical models [2], and evolutionary algorithms [8].

KanbanSimTM presented in: [18] was developed by PMC. KanbanSIMTM facilitates the optimization of Kanban numbers, Kanban container sizes, Kanban reorder points, material handling equipment (e.g., fork lift trucks and Kanban train configurations), etc. It also captures variations in inventory levels at assembly stations, shows the dynamic traffic interactions, schedules material handling equipment, and predicts the impact of material handling on system throughput.

Qi Hao and Weimimng Shen [19] propose a hybrid simulation approach, using both discrete event and agent-based technologies, to model complex material handling processes in an assembly line. Material Kanbans are introduced as an effective means to control and balance the physical material/part flow in the plant floor. An agent-based simulation prototype is implemented using Anylogic™. L.S. Chai [20] proposes a multi agent- based systems architecture of e-based supply chain Kanban integration. Agent coordination using an extended contract net protocol is adopted. The proposed heuristic and programming models for the Kanban planning and coordination capable of handling bidder selection and order allocation for delivery schedules will be demonstrated to work effectively under most supply chain planning scenarios.

However, the models used in the above Kanban simulation researches are all quite simple and far from reality. Different simulation experiments lead to different conclusions because of constrained experimental assumptions.

2 Problem Description and Background

The enterprise analyzed is working in automotive industry specializing in production of vehicles' interiors. It was founded in 2003. It currently employs about 700 people. Its customers are well known automotive concerns, including Ford, Volvo, Opel, Audi, VW. All the products manufactured are exported.

The enterprise is a part of a concern, which is one of the world's leader in the branch. The concern products include: door panels, dashboards, headrests, seats, exhaust systems, rear shelves, bumpers. The enterprises analyzed products are door panels and dashboards.

In the enterprises analyzed, production processes are organized and managed as pull processes. The analyzed pull system is realized in dashboard of vehicle X production process (see photo 1).

Photo. 1 Dashboard after assembly in a vehicle type analyzed

Production process of a dashboard starts with providing granulate to injection department. Finished elements (left and right carrier, left and right carrier reinforcement, upper and lower air channel, left and right gloves box) together with kanban cards, are passed to reference warehouse of department 1 (W1).

After injection stage is finished, the elements are passed to assembly department (W2). The first stage realized in the assembly department is foaming. Foamed elements surface is the one presented in a finished car. After foaming the element, together with kanban cards, are passed to foaming process reference warehouse. In the next stage a carrier is passed to a miller and processed, then, together with kanban card, passed to a warehouse after milling process. At milling stage, depending on order, there is an option of processing with a hole for laser projector cutting procedure.

After unnecessary parts/ surfaces are cut out, milled carrier is treated with welding process. The goal of this, including three operations, process is providing a future dashboard with proper mechanical resistance. Assembled dashboard is fixed in a container. After a container is full/ complete it is passed to a logistics service provider. Products leave production line in sequences. The container taken is passed to shipment preparation area together with finished products, and is palletized and waits for shipment there.

Information flow in all the departments is combined with kanban cards loops, the idea of the system is presented in the fig.1 below.

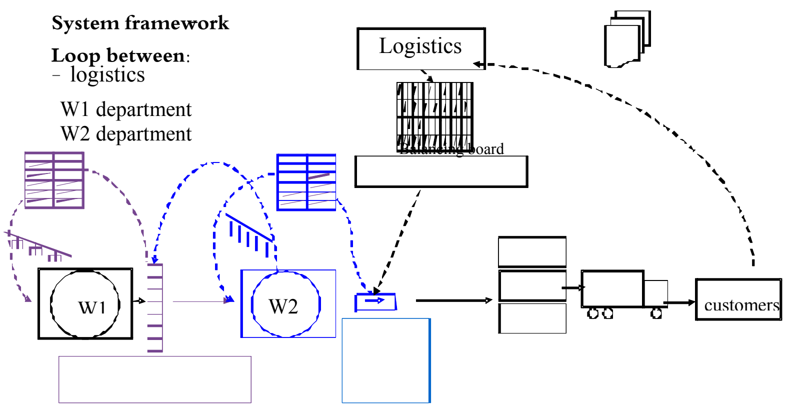


Fig. 1 General scheme for information flows in pull system of project analyzed

Disposer receives information from customers via XPPS system (a software enabling data flows between production facilities, in order to simplify order flows organization, as well as organization of material and stock flows), concerning demand for the next three months in one-day time horizon, then processes it to identify those most important, referring to products ordered for the next few days.

The next step is definition of a production plan for the next day. Production plan defined is represented by a set of logistics kanban cards, and passed to a balance table. Production is performed according to kanban cards withdrew from a balancing table at predefined time (time range: 15 minutes). Every 15 minutes information specifying what product is needed comes out from balancing table and is passed to the assembly line. Specified products are assembled, an employee takes sub-products necessary (WIP – Work in Progress) from welding warehouse, kanban card is taken from the piece chosen and is passed to a box in which predefined production lots are collected. After the lot is collected kanban card is passed to production queue (kanban cards series are put there in predefined order) of welding process. Sub-products in milling and foaming processes are dealt with in a similar way – sub-products are taken, a production lot of cards is collected, and after it is complete it is passed to waiting queue as presented in the figure below (Figure 2). The same scheme is used in injection department – using data concerning backups from warehouses a series is collected, but because of longer set-up time (tools exchange), production series lead time in W1 is longer than production series lead time in W2.

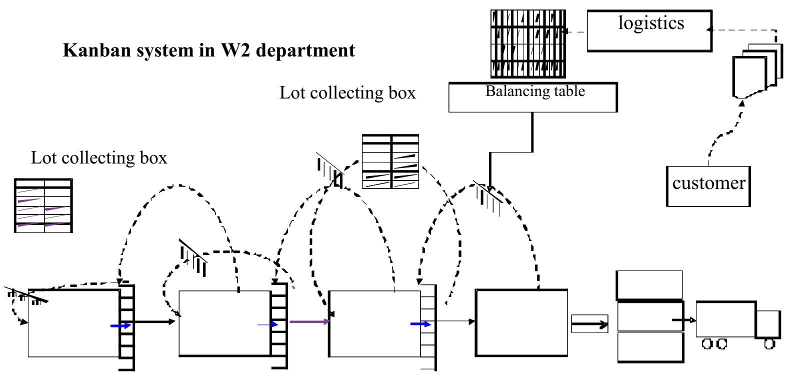


Fig. 2 Information flows in assembly department scheme

The serious problem for the system functioning is the fact, that customers are able to change their orders for the next day, though requiring one week of frozen time horizon is a practice common in production planning. In the case analyzed customers are entitled to changes so the situation in which the changes are quite big are possible to happen. The examples of changes in customers' orders are presented in the table 1.

Table 1 Examples Of Changes In Reference Order Size

Reference	Quantity day1	Quantity day 2	Var %
A	86	38	126%
B	75	58	29%
C	30	24	25%
D	40	50	25%
E	64	75	17%
F	20	23	15%
G	14	68	386%
H	26	18	44%
Sum	355	354	X

Such variance in customers' orders has a substantial influence on pull systems – what happens when customers change their orders so drastically? The following disturbances may appear:

- Percentage of each reference in total references number changes, which results in changes of number of kanban cards in each loop,
- Number of references in warehouse needs to be calculated one more time,
- If the number of orders for reference rarely used so far increases substantially, customers may not receive assortment ordered,
- Large variances in plans destabilize productions process.

3 Results and Conclusions

The conclusion from the analysis presented above is that in the system functioning in the enterprise, a new solution, that would eliminate or at least decrease variations in customers' orders, is required. The solution suggested is reference warehouse in logistics area presented in the fig.3. The warehouse should include all the references, and the number of each reference should be calculated basing on forecasted variation of orders, in three months time horizon. The system with the warehouse should work according to the scheme presented – basing on the information in a balance table:

- Balance production lots should be passed to assembly department,
- References with large variations are taken from a warehouse (feedback information concerning replenishment of elements taken from a warehouse is generated and passed to an assembly waiting queue to rebuilt warehouse stock level in predefined series),
- Assembly department generates products using information from balance cards (smoothed production) and feedback information from warehouse including predefined, collected lots.

After all ordered elements are produced, they are divided into two groups. One of them is passed to shipment preparation area, the other used to rebuilt warehouse. The scheme of the process described above is presented below, with use of the agent system and includes the following agents:

Balancing agent (coordinator): is responsible for storing information coming from logistics agent, assembly agent and warehousing agent, it supports warehousing agent providing it with information concerning stock levels.

Distribution agent: is responsible for collecting customers' orders, it communicates with logistics agent to provide information including customers needs and assembly agent with required number of products to be manufactured.

Assembly agent: controls assembly process (releases orders and controls them).

Warehousing agent: controls warehouse, stock levels.

Logistics agent: responsible for shipments planning and other logistics activities.

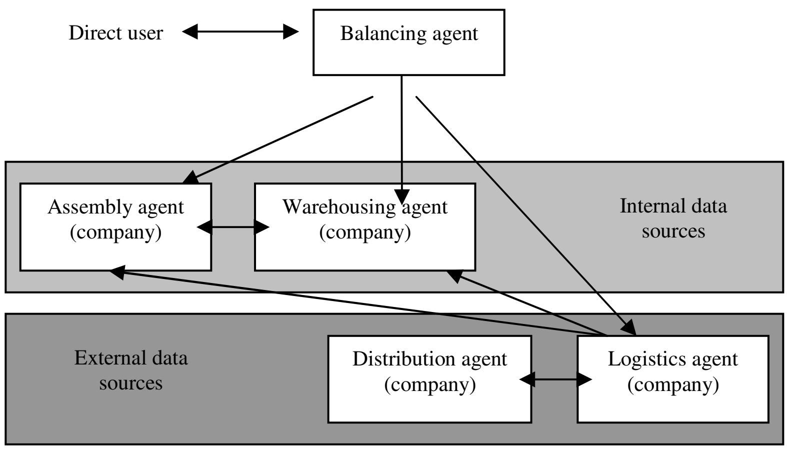


Fig. 3 Allocation of agents in information flows – agent-based system model

The warehouse presented above accumulates sudden increases of orders in all references which enables better production planning. Basic benefits emerging from this solution application are the following:

- Stabilization of material and information flows = stable process,
- Repeatable production plans in following shifts, customers variations are divided into several production shifts,
- Smoothed production plans enable work with all the references in a warehouse, which paradoxically helps in inter-operational stock level reduction and decreases lead time,
- Frequent set-up of inter-operational warehouses in unnecessary, and consequently kanban loops do not have to be frequently adapted,

- Risk of situation in which customers cannot receive ordered parts is decreased,
- Work with a warehouse provides better fitness of manpower employed to operate production lines to workload and products taking pace (keeping overstock is not necessary),
- Warehouse itself is a security tool, providing safety in case of a failure or any other emergency that may occur.

References

1. Lin, G.Y., Solberg, J.J.: An agent-based flexible routing manufacturing control simulation system. In: Proceedings of the winter simulation conference, Lake Buena Vista, FL (1994)
2. Martins, K., Lewandrowski, U.: Inventory safety stocks of Kanban control systems. Prod. Planning Control 10(6), 520–529 (1999)
3. Shen, W., Norrie, D.H.: Agent-based systems for intelligent manufacturing: a state-of-the-art survey. Knowl. Inf. Syst. 1(2), 129–156 (1999)
4. Tommelein, I.D., Weissenberger, M.: More Just-in-Time: location of buffers in structural steel supply and construction processes. In: Proceedings IGLC-7, University of California, Berkeley, CA, USA, July 26-28 (1999)
5. Schonberger, R.J.: The Kanban System. In: Voss, C.A. (ed.) Just-In-Time Manufacture, pp. 59–71. IFS Ltd., UK (1987)
6. Monden, Y.: Toyota Production System. Institute of Industrial Engineers, Norcross (1983)
7. Krishnamurthy, A., Suri, R., Vernon, M.: Re-examining the performance of MRP and kanban material control strategies for multi-product flexible manufacturing systems. Int. J. Flex Manuf. Syst. 16(2), 123–150 (2004)
8. Kochel, P., Nieländer, U.: Kanban optimization by simulation and evolution. Produc. Planning Control 13(8), 725–734 (2002)
9. Gupta, S.M., Yay, A.-T.: The effect of sudden material handling system breakdown on the performance of a JIT system. Int. J. Produc. Res. 36(7), 1935–1960 (1998)
10. Kochel, P., Nieländer, U., Sturm, M.: KASIMIR—object-oriented KAnban SIMulation Imaging Reality. Research report, Chemnitzer Informatik-Berichte CSR-01-03. Chemnitz University of Technology, Chemnitz (2001)
11. Andijani, A.A.: A multi-criterion approach for Kanban allocations. Omega Int. J. Manage Sci. 26, 483–493 (1998)
12. Berkley, B.J., Kiran, A.S.: A simulation study of sequencing rules in a Kanban-controlled flow shop. Decis. Sci. 22, 559–582 (1991)
13. Hum, S.H., Lee, C.K.: JIT scheduling rules: a simulation evaluation. Omega Int. J. Manage Sci. 26, 381–395 (1998)
14. Berkley, B.J.: Effect of buffer capacity and sequencing rules on singlecard Kanban system performance. Int. J. Produc. Res. 31, 2875–2893 (1993)
15. Savsar, M.: Effects of Kanban withdrawal policies and other factors on the performance of JIT systems: a simulation study. Int. J. Produc. Res. 34, 2879–2899 (1996)
16. Karaesmen, F., Dallery, Y.: A performance comparison of pull type control mechanisms for multi-stage production control. Int. J. Produc. Econ. 68, 59–71 (2000)

17. Yavuz, I.H., Satir, A.: A Kanban-based simulation study of a mixed model Just-in-Time manufacturing line. Int. J. Produc. Res. 33, 1027–1048 (1995)
18. KanbanSIM$_{TM}$, PMC, `http://www.pmcorp.com/kanbanSIM.shTMS`
19. Hao, Q., Shen, W.: Implementing a hybrid simulation model for a Kanban-based material handling system. Robotics and Computer-Integrated Manufacturing 24, 635–646 (2008)
20. Chai, L.S.: e-based inter-enterprise supply chain Kanban for demand and order fulfilment management. SIMTech technical reports (STR_V8_N4_09_POM) 8(4) (October-December 2007)

Supply Chain with Reverse Flows- Benefiting from Recycled Resources

Konrad Fuks, Paulina Golinska, and Agnieszka Stachowiak

Abstract. The research to be presented is a case study introduced to analyze relations between a company and its suppliers, as well as consequences of decisions concerning suppliers selection. The consequences analysis is based on simulation in which outcomes are average costs of resources depending on suppliers selected. The results of the analysis can be helpful for other companies when selecting suppliers, as it shows relations between suppliers, resources they provide, and final product, its quality and cost. An important aspect of the analysis and the case study presented is using reverse material flows, as alternative sources the company uses are recycling companies. Hence, except from economical, ecological point of view is given.

Keywords: cooperative purchasing, logistics services, e-supply chain, semantic web, multi-agent system.

1 Introduction-Project Background

Ecological issues are nowadays of growing importance, not only in waste management area, but also in all the supply chain, starting from supplies, through manufacturing, distribution, exploitation and reverse flows management. Using recycled materials from reverse flows is ecological and, what is important for each company, economical. That is the reason why more and more companies strive for benefiting from reverse material flows use. They buy materials from recycling companies, which collect and process it, or buy used products from their customers, trying to close the supply chain loop. Diversification of suppliers,

---

Konrad Fuks
Poznan University of Economics
Al. Niepodleglosci 10, 61875 Poznan, Poland
e-mail: Konrad.fuks@ue.poznan.pl

Paulina Golinska · Agnieszka Stachowiak
Poznan University of Economics
Strzelecka 11, 60965 Poznan, Poland
e-mail: paulina.golinska@put.poznan.pl, a_stachowiak@wp.pl

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 277–284.
springerlink.com

especially for raw materials of strategic meaning, is crucial for company's seamless production and material safety in dynamic, unpredictable environment. In this paper authors examine the potential of agent technology for solving problems in area of materials management, where both forward (supplies) and reverse (re –supplies) flows of materials are applicable. Closed loop supply chain is a promising field of application because of its inherent complexity, which is characterized by [4,5]:

- heterogeneous environments,
- open and dynamic structure,
- interoperability of components,
- distributed processes,
- scalability.

The effectiveness of the supply chain performance might be measured by its ability to provide customers with due-to-date deliveries of final goods and possibility of effortless collection of the used products when they're obsolescent (reverse logistics). Agent-based system architecture provides an interesting perspective for integration of material flows originating from dispersed locations in the supply chain, due the fact that agents are able to generate, process, store, filter, correlate, broadcast, and route information for real-time coordination and decision making. Agents have the following characteristic [13]: reactive, pro-activeness, social ability. The above mentioned characteristics suits well to requirement for reverse material flow management like common goal orientation, ability to interact with other supply chain participants and reactivity to disturbances appearing in material flows. Agent systems perfectly suit the demands for global flexibility, cooperation and at the same time, local autonomy. Compared to existing SCM systems, allows the successful integration of both inter- and intra- organizational planning.

The research is to examine potential of agent technology for company trying to gain independence from its previous supplier (a monopolist) and to be more economic and ecologic at the same time. The research introduces the case of a company trying to maximize supplies flows from secondary sources. The problem is analyzed and solved with simulation using multi-agent approach. The first section introduces literature background concerning reverse flows in supply chains. The second introduces the problem and the data collected. The third presents the model used in problem solution and the goal of the simulation, the next briefly describes simulation environment used. The results are to be described and analyzed in the fifth section, summarized with conclusions.

2 Reverse Flows in Supply Chain and Their Influence on Logistics Operations

Reverse flows and reverse logistics are now gaining more and more attention, as ecological approach to business has become not only a fashion, but simply a necessity. Benefits from integration and supply chain strategy were presented in numerous works on the subject. Supply chains including reverse flows and closed-loop supply chain are also subject to research and publications, including Blumberg *et al* [1], Fleishmann *et al* [3], Guide *et al* [6], Golinska *et al* [4,5], Rogers *et al* [9], Seitz *et al* [10] and others. According to the approach presented in the literature, ideal closed loop supply chain can be defined as zero-waste supply chain that completely reuses and recycles all materials. Such situation is not common, and disposal takes place, but the amount of goods being disposed should be minimized.

Reverse flows in companies generally include [3]: end-of-life returns, warranty returns, production scrap and by products, reusable packing material. The flows can differ with quality and, consequently, value, with end-of-life returns and warranty returns being the most valuable.

The reasons why companies try to close their supply chain are legislative (strict provisions concerning waste disposal and using non-renewable resources), marketing (creating ecological image of a company), economics (cheaper resources that can be used without influencing quality of a final product, subventions, preferential tax rates).

There are several approaches to closed loop supply chain modeling, including the one presented by Krikke [8], according to which, there are three basic models of closed loop supply chain: control, responsive and efficient. The control supply chain deals mostly with reverse flows coming from the market as end-of-life returns. The responsive closed loop supply chain deals with end-of-life returns too, however it also includes commercial returns and warranty returns. The efficient closed loop supply chains deals with production scrap, by products and reusable packing materials.

2.1 Problem Definition

The aim of the simulation is to examine distributed sourcing and re-sourcing strategy in an aluminum Christmas decoration manufacturer of a medium size. The optimaziation task is understood as a maximization of returns (recycled resources) and minimization of raw materials flows.

The analyzed company uses one production line, has no warehouse to store raw materials or finished goods. The production is seasonal, run only for six weeks a year. There is one strategic raw material used – aluminum, the other materials are of a lesser importance and are not a subject of the analysis. The company gets aluminum from aluminum works or from recycling companies collecting and processing aluminum.

The material supplied by the aluminum works is of high quality and relatively expensive, however the supplies are provided with some constraints, which are fixed minimum order quantity and fixed order cycle.

The material supplied by the recycling companies is relatively cheap but of poor quality and therefore needs to be additionally processed (which generates extra costs), delivery cycle is unpredictable, order quantity varies on account of availability of material.

The supply chain to be analyzed is not a closed loop supply chain, as it uses material coming from other companies, and not only relies on supplies coming from its customers. However, it wants to benefit from reverse flows, to be ecological, to be economic, it generally purchases production scrap and by products, used goods and reusable packaging material.

2.2 Data and the Identified Constrains

Data and constraints are grouped respectively to the area they concern.

Aluminum works characteristics
Number of aluminum works = 1-3
Aluminum price = 18.000-20.000$ per 1ton
Delivery cycle (from aluminum works) = 7 days
Minimum order size (from aluminum works) = 40-60% of demand (randomly chosen for each works)
Recycling companies characteristics
Number of recycled aluminum providers = 20-25
Recycled aluminum price = 5.000 – 6.500$ per 1ton
Recycled aluminum processing costs = 6.500 – 8.000$ per 1ton
Delivery cycle (from recycling companies – includes processing cycle) = 3-14 days
Minimum order size (from recycling companies) = 100kg
Supplies = minimum 0.5% of weeks demand, maximum 10%-20% (number chosen randomly) of weeks demand
Christmas decoration manufacturer characteristics
Demand = 4-6 tons
Demand per week = 12.5% of total demand in weeks 1, 2, 3 and 4, and 25% of total demand in weeks 5 and 6
Transaction cost = 5-10% of price (randomly set at the beginning of the week)
Storing cost = 20% of price

The number of aluminum works and aluminum recycling companies is set after local market analysis. The prices are set as an average, basing on data from aluminum stock, collected between 01-09 January 2009. The cost of processing is set basing on opinions from recycling experts. The demand, delivery cycles, minimum order sizes are data from company's annual reports (historical data) Quality dependent on price (the lower price, the higher processing cost and waste amount). At the beginning of the week the price, recycling companies supplies amount and delivery cost is known.

3 Model

The relations between analyzed company (Christmas decoration manufacturer) and its environment, comprising of its suppliers, are presented in the model below (Fig.1). The company introduces mix procurement approach combining both raw materials (from aluminum works) and recycled material (from recycling companies). Companies collect waste aluminum and process it, processed material is sold to the decorations manufacturer. The amounts supplied by the aluminum works are greater than those supplied by the recycling companies. Quality of waste material influences processing parameters (cost and efficiency), processed aluminum is the same quality as this bought from the works.

Company's supplies can be described as follows:

$$S = \sum_{i=1}^{2} AW_i \bullet S_{RM} + \sum_{j=5}^{15} RC_j \bullet S_{PM} \tag{1}$$

Where:
i – number of aluminum works
j – number of recycling companies
AW – aluminum works
S_{RM} – supplies of raw material
RC – recycling company
S_{PM} – supplies of processed material

The goal of the company and the simulation to be run is to maximize supplies of processed material:

$$\sum_{j=5}^{15} RC_j \bullet S_{PM} \rightarrow \max \tag{2}$$

Maximizing supplies of processed material is both economical and ecological, but one has to deal with uncertainty of delivery cycle and quantities of material available. The simulation is to present relations between the number of aluminum works, recycling companies and amounts of waste and processed aluminum available.

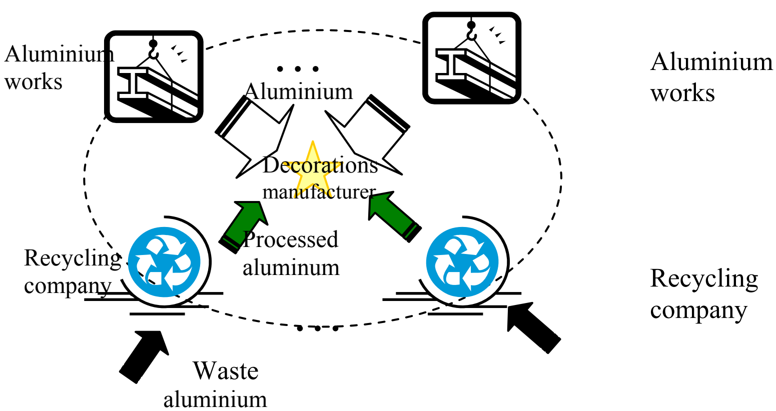


Fig. 1 The company and its suppliers

4 Simulation

NetLogo is a programmable modeling environment for simulating natural and social phenomena. NetLogo was authored by Uri Wilensky in 1999 and is under continuous development at the Center for Connected Learning at Northwestern University. It is the next generation of the series of multi-agent modeling languages that started with StarLogo. The environment allows to give instructions to a lot of independent agents which can interact with each other and perform multiple tasks [see 11,12].

The results obtained with the simulation include the number of recycled aluminum providers, the number of aluminum works, average cost of material and when, in manufacturing cycle, demand was met.

The results to be analyzed obtained with the simulation are presented in the Table 1 and in the Figure 2. The results include the number of recycled aluminum providers and minimum and maximum average cost of material that can be achieved in the simulated market environment (factors influencing the price/ cost are the number of recycled aluminum providers and the number of aluminum works).

Table 1 The relation between the number of resources providers and the cost

number of recycled aluminum providers	min av cost	max av cost
1	17,24	19,67
2	16,51	19,51
3	15,85	19,37
4	15,02	19
5	14,26	18,04
6	13,94	17,68
7	13,87	17,41
8	11,86	16,79
9	11,87	16,87
10	11,88	16,22
11	12,05	15,5

The results can also be presented as in the Figure 2.

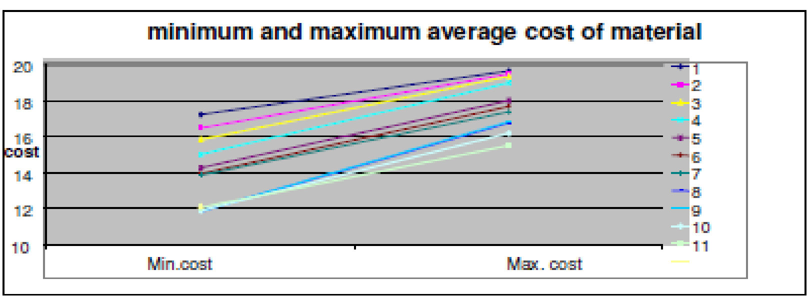


Fig. 2 The relation between the number of resources providers and the cost

Analysis of the results clearly shows that the best procurement strategy for the manufacturer is to cooperate with eight recycled aluminum providers and to quit the cooperation with aluminum works. Competitive position of aluminum works is very high, especially comparing to market position and demand of the company analyzed. It makes negotiation and contract conditions customization difficult, and as a result increases material cost. On the other hand recycled aluminum providers are companies of a size and position similar to their purchaser, and being smaller they are more flexible and responsive. In the environment simulated eight recycled aluminum providers is enough to satisfy company's demand in acceptable time, provides safety necessary when dealing with reverse flows, which are unpredictable and unstable (the number of goods returned to recycling companies depends only on people using them, and their decisions cannot be easily forecasted, as well as quality of material returned). Number of recycled aluminum providers greater than eight seems inefficient, as minimum and maximum average cost increases. Dealing with too many suppliers may be risky as it weakens the relations, partnership in the supply chain.

5 Conclusions

The research presented was supposed to show importance of suppliers selection and financial consequences of decisions concerning suppliers. The case study presented is quite simple as it focuses on one resource only – nevertheless it is a raw material crucial for production. Aluminum is an example of rare and expensive resource and its reverse flows are well organized on the market analyzed (comparing to other resources that can be recycled). Manufacturing process employed is simple and its analysis in terms of time, cost and quality is possible. Even though simple, the case present shows that the number of supplier influences the cost and consequently, the price of ready to sell products. The simulation also proves that ecological sometimes also means economic, as reusing material returned showed to be cheaper than using raw material, even though some additional costs connected with processing and recycling were necessary. In the case analyzed suppliers – and resources – influence cost, as it was discussed above, as well as quality and time-to-market. Cost, quality and time-to-market depend greatly on resources provided as they determine the manufacturing process – its stages, resources to be used: technology, machines, materials, manpower, time etc. For the company analyzed in the research quality is the least important, as Christmas decorations generally (with some exceptions) are cheap, popular products, used once in a year, they are not exposed to difficult conditions, they do not need to be durable. The most important for the value of the final product analyzed is time-to-market, as it is seasonal, and after high season is looses most of its value. Time-to-market is influenced with material to be used as it determines delivery and processing cycles, so the decisions concerning resources and their supplier influence this aspect and value of a product as well. The further research should examine the optimization of multi- resource materials requirements for particular production purpose.

References

1. Ardissono, L., Petrone, G., Segnan, M.: A conversational approach to the interaction with Web Services. Computational Intelligence 20, 693–709 (2004)
2. Blumberg Donald, F.: Strategic Examination of Reverse Logistics and Repair Service Requirements, Needs, Market Size and Opportunities. Journal of Business Logistics 20(2), 141–159 (1999)
3. Fleischmann, F., Beullens, P., Bloemhof-Ruwaard, J.M., van Wassenhove, L.N.: The Impact of Product Recovery on Logistics Network Design. Production and Operations Management 10(2), 156–173 (2001)
4. Fleischmann, M.: Reverse logistics network structures and design, Erasmus Research Institute of Management. In: Guide Jr, V.D.R., Van Wassenhove, L.N. (eds.) Business Aspects of Closed Loop Supply Chain. Springer, Berlin (2003)
5. Golinska, P., Fertsch, M., Gómez, J.M., Oleskow, J.: The concept of closed-loop supply chain integration through agents-based system. In: Information Technologies in Environmental Engineering, ESE. Springer, Heidelberg (2007)
6. Golinska, P.: The concept of an agent-based system for planning of closed loop supplies in manufacturing system. In: Omatu, S., Rocha, M.P., Bravo, J., Fernández, F., Corchado, E., Bustillo, A., Corchado, J.M. (eds.) IWANN 2009. LNCS, vol. 5518, pp. 382–389. Springer, Heidelberg (2009)
7. Guide Jr., V.D.R., Van Wassenhove, L.N. (eds.): Business Aspects of Closed Loop Supply Chain. Springer, Berlin (2003)
8. Krikke, H., Pappis, C.P., Tsoulfas, G.T., Bloemhof-Ruwaard, J.B.: Design principles for closed loop supply chains: optimizing economic, logistic and environmental performance. Erasmus Research Institute of Management, Report series research in management (2001)
9. Krikke, H., le Blanc, I., van de Velde, S.: Product modularity and the design of closed-loop supply chains. California Management Review 46(2), 23–39 (2004)
10. Rogers, D.S., Tibben-Lembkers, R.: Going Backwards: Reverse Logistics Trends and Practices, Pittsburgh, Reverse Logistics Executive Council, 1–137 (1999)
11. Seitz, M.A., Peattie, K.: Meeting the closed-loop challenge: the case of remanufacturing. California Management Review 46(2), 74–90 (2004)
12. Tisue, S., Wilensky, U.: NetLogo: Design and Implementation of a Multi-Agent Modeling Environment. Presented at SwarmFest, Ann Arbor, May 9–11 (2004), http://www.ccl.sesp.northwestern.edu/papers/netlogo-swarmfest2004.pdf/
13. Wilensky, U.: NetLogo itself, NetLogo, Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL (1999), http://ccl.northwestern.edu/netlogo/
14. Jennings, N.R., Wooldridge, M.J.: Applications of Intelligent Agents. In: Agent Technology: Foundations, Applications, and Markets. Springer, Heidelberg (1998)

e-Wedding Based on Multi-agent System

Kobkul Kularbphettong, Gareth Clayton, and Phayung Meesad

Abstract. Multi-agent system is continuously utilized in e-Business that want to improve responsiveness and efficiency of systems. In this paper, we propose multi-agent-system and various techniques that are Web service, ontology, and data mining techniques. The multi agent system, which constitutes the backbone of the framework, connects these pieces together and makes them perform properly. JADE is the MAS platform implemented this project. JADE is quite easy to learn and use. Moreover, it supports many agent approaches such as agent communication, protocol, behavior and ontology. This framework has been experimented and evaluated in the realization of a simple, but realistic, prototype of an e-Wedding system. The results, though still preliminary, are quite encouraging.

Keywords: multi-agent system, Web services, ontology, data mining techniques, JADE, e-Wedding.

1 Introduction

Nowadays, the explosion of Internet and the grown-up of e-Commerce have rapidly changed the traditional business operation. Electronic Commerce, commonly known as e-commerce or eCommerce is the development and deployment technology of the commercial transactions electronically so as to meet the ever-growing demands of the modern life. Based on web-based business, the wedding business is one of the important businesses that it be huge and still rapidly expanding. Booming of wedding business has also propelled the enhance of hotels, wedding studios, car hiring companies, flower shops, music, travel agencies, and even media businesses. Due to overwhelming information widely spread on internet and real world environments, it is very time consuming for couple to search appropriate information. Moreover, almost successful e-Commerce systems are still handled by humans to make the important decisions. Therefore, with vastly developed advance mechanisms, in this paper we propose the framework of e-Wedding business applied various approaches like multi-agent, web services, ontology, and data mining techniques.

The remainder of this paper is organized as follows. Section 2 discusses about related literatures and research works. Section 3 presents the methodologies used in this work. Section 4 implements the purposed model with a multi-agent

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 285–293.
springerlink.com

framework. This prototype demonstrates the whole idea of adapt multi-agent in e-Wedding. Finally, we conclude the paper with future research issues in section 5.

2 Related Works

A literature search shows that most of the related researches have been deployed multi-agent to develop e-Commerce in various techniques by following this:

According to [1], they showed a prototype of the system using the JADE platform in the context of travel industry. Furthermore, other research works show that agent technologies are deployed as a significant tool for developing e-Commerce applications [2]-[7]. Hence, multi-agent technology, a promising approach, trends to handle internet transaction for customers.

Moreover, other researchers propose an agent-based framework representing in various ways.[8]-[11],[14] For instance, A.Negri, A. Poggi, M. Tomaiuolo, P. Turci [12] integrated the agent technology with other key emerging technologies like semantic Web, Web service, rule engine and workflow technologies and KODAMA [13] is another system based on a multi-agent system and leveraged the Semantic Web services.

From previous literature works, it appears that there are many research studies exploiting various techniques blended with multi-agent technology. Therefore, in order to success on e-Commerce, agent should have abilities to perform as a behalf of user to handle with business tasks such as planning, reasoning and learning. Data mining techniques is the important way to make a reason for agent under uncertainty and with incomplete information situations.

3 The Methodologies

3.1 Multi-agent Systems

Agent is the software program that enables to autonomous action in some environment so as to meet its design objectives. According to N. R. Jennings and M. Wooldridge [15], the essential characters of each agent are following: reactive, pro-active, autonomous, object-oriented and social ability. Each agent can play as a behalf of the user and execute the particular task. However, in open and dynamic environment like internet, a multi-agent system is one of the important means to help reduce cost, increase efficiency, reduce errors and achieve optimal deal.

There are two issues related to the design of MAS: Agent Communication Language and agent development platform. The former concerns with the message interchange between different agent such as KQML, and FIPA ACL. The latter is related with the platform development to provide an effective framework, such as IBM Aglets, ObjectSpace Voyager and etc, for the dispatching, communications, and management of multiple agents in the open and dynamic environment. For this proposed project, JADE (Java Agent Development Framework) will be deployed as the prototype development tool.

JADE (Java Agent Development Framework) is a software environment fully implemented in JAVA language aiming at the development of multi-agent systems that comply with FIPA specifications [20]. The goal of JADE is to simplify development while ensuring standard compliance through a comprehensive set of system services and agents. Each running instance of the JADE runtime environment is called a container as it can contain several agents. The set of active containers is called a platform. A single special container must always be active in a platform and all other containers register with it as soon as they start.

3.2 Web Services

The W3C Web Services Architecture Working Group defines a Web service as:

"*A software application identified by an URI, whose interfaces and bindings are capable of being defined, described and discovered as XML artifacts. A Web service supports direct interactions with other software agents using XML-based messages exchanged via Internet-based protocols.*"[17].

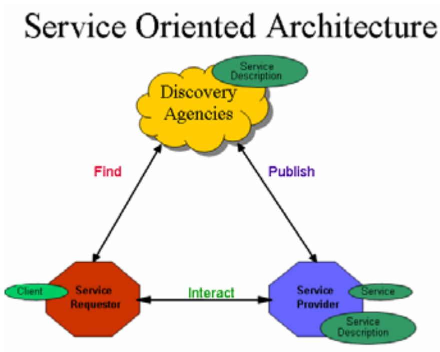


Fig. 1 The Web Service Architecture [18]

As Figure 1 represents the Web Service Architecture is included 3 components: Discovery agencies, service requesters and service providers. The Web service technology, allows uniform access via Web standards to software components residing on various platforms and written in different programming languages. As a result, software components providing a variety of functionalities (ranging from currency conversion to flight booking) are now accessible via the Web. Indeed, Web service technology has introduced a new abstraction layer over and a radically new architecture for software. From a business perspective, Web services often correspond to business services and thus the compositionality paradigm that underlies the Web service technology allows composing existing business services into new and more complex services.

3.3 Ontology

Ontology is an important composition in the communication language. Ontology is an agreement about a shared conceptualization, which includes frameworks for modeling domain knowledge and agreements about the representation of particular domain theories, often captured in some form of a semantic web formally. Its aim is representing the shareable conceptual model in formalized specification [8].

There are two kinds of ontology in the communication model, kqml-ontology and negotiation-ontology. Kqml ontology has been defined formally. We can find the OWL version from The DARPA Agent Markup Language web. Negotiation ontology is based on the idea that there are some general concepts that are present in any negotiation, and builds on finding commonalities across different negotiation protocols [9].

3.4 Data Mining Techniques

Data mining has matured as a field of basic and applied research in computer science in general and e-Commerce in particular [19]. The success of DM is

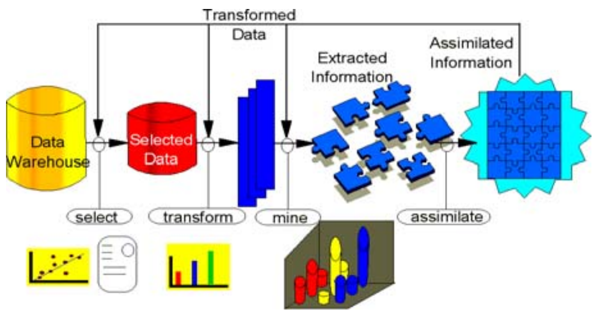


Fig. 2 The Data Mining process [21]

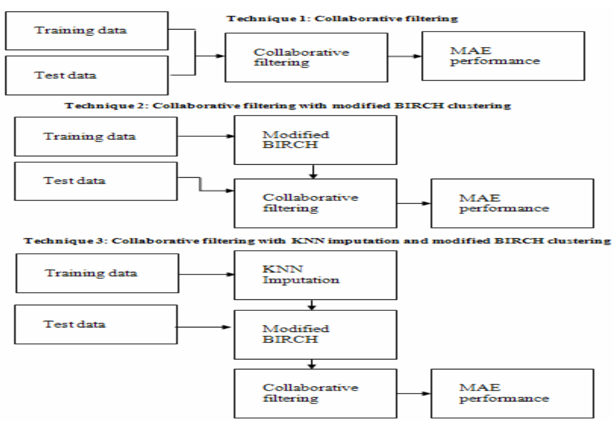


Fig. 3 The flowchart of data mining in the e-Wedding system

driven to a very large extent by the following factors: availability of data with rich descriptions, availability of a large volume of data, reliability of the data available, ease of quantification of the return on investment (ROI) in DM and ease of interfacing with legacy systems [20]. In Figure 2 explained the data mining process, from this project, we used Collaborative filtering, Collaborative filtering with modified BIRCH clustering, Collaborative filtering with KNN imputation and modified BIRCH clustering techniques as showed in figure 3.

4 The Proposed Architecture of e-Wedding

Today the wedding business is increasingly becoming growth. With its boom, it is effect to other business like Hotel, wedding studio, car hiring companies, flower shops, music, travel agencies, and even media businesses. Due to the explosion of internet, couples use the wedding portal sites as a medium to search needed information. However, it takes time consuming because of the flood with information available on internet. Couples must take time to looking for wedding packages and related services, depended on information of wedding business agencies. Further, couples make their decision based on comparing various wedding packages. The wedding package is composed of hotel wedding package, wedding studio package, music, floral decoration, card & gift agencies and etc.

The proposed e-Wedding framework is a multi-agent system designed to couple with optimum wedding packages for the couples relying on their preferences and our system is divided to be two parts. First, it concerns about a recommendation system to couples to find appropriated wedding packages and the final takes responsibility to bargain with wedding business agencies in order to get the optimal solution. From figure 1 representing the purposed architecture of the e-Wedding system, this framework concerns about four main aspects: multi-agent system, Web services, ontology and data mining techniques.

In the multi-agent system, each agent is autonomous to be able to make decisions and act proactively. Agents can communicate, exchange knowledge, collaborate or negotiate with each other, to efficiently achieve the common goal. They receive and process users' requirements, make use of user's preference, perform as a mediator of services business. They select and contact with appropriate wedding business agencies like hotels, wedding studio, and so on through Web services and give couples the optimal result. Web services define, provide related services, and interact with negotiator agents and ontology is the meaningful data which can be directly accessed by agents or people through Web.

Each agent is identified to response a specific task. An interface agent is designed for assistant couples and acts as a behalf of couple when using e-Wedding. A couple fills their preference with an interface agent and passes though a preparation agent. Moreover, the interface agent observes a couple, adapts preference based on couple's requirement, and returns the result to a couple. A preparation agent responds to calculate couples' budget, searches appropriated wedding business and passes though a manager agent. A manager agent is response for communication with a negotiator agent to send user information, to receive wedding packages, to look for and estimate suitable wedding packages based on user's profile and preference. Furthermore, it takes responsibility for contact with interface agents to purpose wedding packages suitable with their user. A negotiation agent carries out user's information and connects to web services so as to get information about wedding packages. Moreover it persuades its offer to wedding suppliers, evaluates value by compare information with each other and sends to a manager agent in order to evaluate the optimal solution and return the solution to an interface agent. And data mining agents in this system are separated to be three parts: First, a preprocessing Agent, It prepares data for mining to perform the necessary data cleansing, transforming, and preparing data depended on specific data mining techniques. Second, mining agents implement specific data mining algorithms. In this paper as Figure 2, we used Collaborative filtering, Collaborative filtering with modified BIRCH clustering, Collaborative filtering with KNN imputation and modified BIRCH clustering techniques in part of recommendation to search appropriated packaged for couples. Final, auction agent gets the results from mining agents, evaluates the optimum solution and pass to knowledge of system.

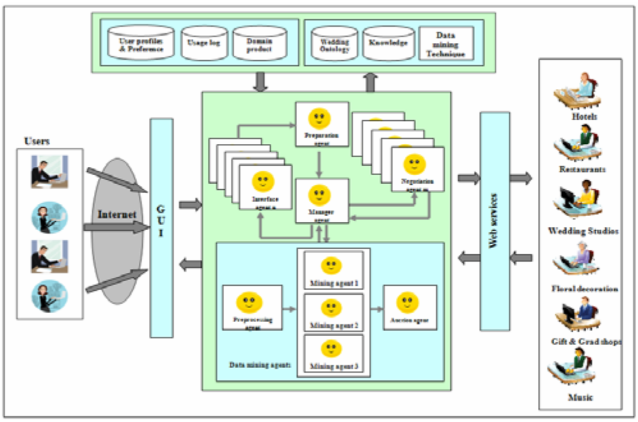


Fig. 4 The purposed architecture of the e-Wedding system

5 Implementation

Our research purpose is to develop a new framework and mechanisms of e-Wedding based on a multi-agent system. In order to demonstrate the effectiveness of our framework and mechanisms, we develop and test a prototype of the e-Wedding.

In the part of user, user can get to our web page, register to be a member and fill in he/she preferences that are included engagement date, engagement place, wedding date, wedding place, expected guest in engagement day and expected guest in wedding day and their estimated budget. After users key in their preference, agents will prepare data, calculate their preferences, connect with Web Service to find suppliers that propose their product to company, contact with

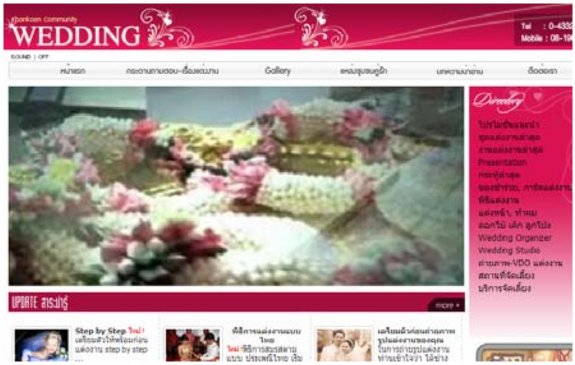


Fig. 5 The prototype of e-Wedding system

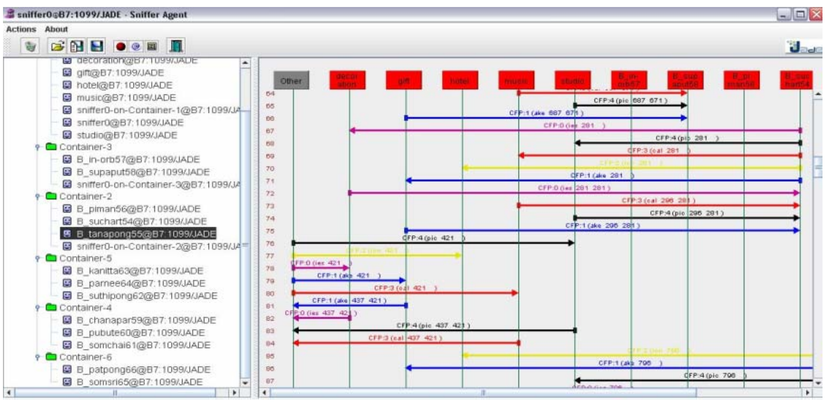


Fig. 6 The operation of JADE

supplier agent to find available products, find optimized packages that suitable with user preferences, purpose optimized packages to customers, and etc. In the part of find optimized packages, we used mean square error sense.

$$y = p_{1 \times m} x_{m \times 1} \quad (1)$$

y = estimated budget of customer
p = real price that we get from our suppliers
x = expected profit

$mse \rightarrow y = px$ and $|y - \hat{y}|^2 \rightarrow 0$

$\min_x |y - \hat{y}|^2$, so that

$$\min_x |y - p\hat{x}|^2 = \min_x (y^2 - 2yp\hat{x} + (p\hat{x})^2) = \min_x \{-2yp\hat{x} + x^T p^T px\} \quad (2)$$

And Figure 4 showed the result of communication among other agent in e-Wedding System by using JADE.

6 Conclusion and Future Works

In this paper we presented our preliminary ideas of building e-Wedding system based on a multi agent system and various techniques. Our system consists of four main aspects: multi-agent system, Web service, ontology and data mining techniques. In the part of MAS, we have implemented this prototype by using JADE platform. JADE is quite easy to learn and use. Moreover, it supports many agent approaches such as agent communication, protocol, behavior and ontology. As for the future work, we need to explore more reasonable technologies and methods to enhance e-Wedding system and also need to evaluate our system from the users' point of view.

References

[1] Balachandran, B.M., Enkhsaikhan, M.: Developing Multi-agent E-Commerce Applications with JADE. In: Apolloni, B., Howlett, R.J., Jain, L. (eds.) KES 2007, Part III. LNCS (LNAI), vol. 4694, pp. 941–949. Springer, Heidelberg (2007)
[2] Cao, M.-K., Feng, Y.-Q., Wang, C.-Y.: Designing Intelligent Agent for E-Business Oriented Multi-Agent Automated Negotiation. In: Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou, August 18-21 (2005)
[3] Hung, P.C.K., Mao, J.-Y.: Modeling of E-negotiation Activities with Petri Nets. In: HICSS 2002 (2002)
[4] Yuan, Y., Rose, J.B., Archer, N., Suarga, H.: A Web-Based Negotiation Support System. International Journal of Electronic Markets (1998)
[5] Piccinelli, G., Bartolini, C., Preist, C.: E-service composition: supporting dynamic definition of process-oriented negotiation. In: Proc. 12th International Workshop on Database and Expert Systems Applications (DEXA 2001), Munich, Germany, September 2001. IEEE Computer Society, Los Alamitos (2001)
[6] Tiwana, A., Ramesh, B.: E-Services: Operations, Opportunities and Digital Platforms. In: Proceedings of the 34th Annual Hawaii International Conference on System Sciences (HICSS-34), In the e-Services: Models and Methods for Design, Implementation and Delivery" mini-track of the Decision Technologies for Management track (2001)
[7] Akkermans, H.: Intelligent E-Business - From Technology to Value. IEEE Intelligent Systems 16(4), 8–10 (2001)
[8] Gutknecht, O., Ferber, J., Michel, F.: Integrating tools and infrastructures for generic multi-agent systems. In: Proc. of the 5th International Conference on Autonomous Agents, Montreal, Canada (2001)
[9] Hendler, J.: Agents and the semantic web. IEEE Intelligent Systems 16(2), 30–37 (2001)

[10] Bergenti, F., Poggi, A., Tomaiuolo, M., Turci, P.: An Ontology Support for Semantic Aware Agents. In: Kolp, M., Bresciani, P., Henderson-Sellers, B., Winikoff, M. (eds.) AOIS 2005. LNCS (LNAI), vol. 3529, pp. 140–153. Springer, Heidelberg (2006)

[11] Labrou, Y.: Agents and ontologies for e-business. Knowledge Engineering Review 17(1), 81–85 (2002)

[12] Negri, A., Poggi, A., Tomaiuolo, M., Turci, P.: Agents for e-Business Applications. In: International Conference on Autonomous Agents Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems Hakodate, Japan (2006)

[13] Yu, H., Mine, T., Amamiya, M.: A Multi-Agent-Based Semantic Web Service System. In: The International Symposium on Information Science and Electrical Engineering 2003 (ISEE 2003), Fukuoka, Japan, November 13-14, pp. 633–636 (2003)

[14] Yu, H., Mine, T., Amamiya, M.: An Architecture for Personal Semantic Web Information Retrieval System. In: Integrating Web services and Web contents The 2005 IEEE International Conference on Web Services (ICWS 2005), Orando, USA, July 11-15, pp. 329–336 (2005)

Distributed Computing Using RESTful Semantic Web Services

Antonio Garrote Hernández and María N. Moreno García

Abstract. In this article *RESTful semantic web services*, a lightweight alternative for the description of semantic web services, are introduced. The conceptual model for this kind of semantic services is reviewed, showing the relationships between semantic meta-data and formalisms like triple space computing and REST constructs. An extension to the polyadic Pi-Calculus to model composition and coordination of semantic services is proposed The resulting computational model can serve as the basis for the description of agent systems implemented using standard semantic and web technologies. Finally, a sample implementation of a web application built using a composition of RESTful semantic web services is also proposed.

1 Introduction

Web services ultimate goal has always been turning the World Wide Web into a truly distributed computing platform. Advances in web services and service-oriented architectures have provided a set of technologies and standards that can serve as the underlying infrastructure enabling distributed applications in the web. However, these technologies face important difficulties in areas such as the discovering and composition of web services. Another recent evolution of the web, the so called semantic web, has the potential to address these issues.

Semantic web technologies allow the automatic processing of web data. They have evolved in parallel with the spreading of web services architectures. Both technologies can be considered complementary [4]. Research on the convergence between web services and semantic web has produced as a result a number of standard

Antonio Garrote Hernández
Universidad de Salamanca
e-mail: agarrote@usal.es

María N. Moreno García
Universidad de Salamanca
e-mail: mmg@usal.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 295–302.
springerlink.com

proposals for the specification of semantic web services. W3C's standard proposals: Web Services Modeling Ontology (WSMO) [10], OWL-S [11], the Semantic Web Services Framework (SWSF) [2] or WSDL-S [1] are some of the most prominent.

On the other hand, web services standards like WSDL, SOAP, known collectively as WS-* web services, have received harsh criticism among practitioners due to the perceived unnecessary complexity of these standards. This criticism has crystallized into the proposal for Representational State Transfer (REST) [5] service-oriented architecture. REST proposes following HTTP design guidelines in order to achieve simple and scalable web services.

Semantic web services describe ontologies and process models that have been traditionally transformed into WS-* services and messages for execution. Recent research has been made in some proposals, like WSMO, to provide alternative grounding mechanism [9] compatible with REST architectures and *triple spaces*[4].

In this article, practical application of this kind of simple semantic web services, based in the concepts of triple space computing and RESTful architectures is reviewed and a proposal for extending RESTful concepts in a model for semantic web services is discussed.

2 Service Model

The service model for RESTful semantic web services must be described at different levels as shown in the figure 3:

- Concepts and instances of concepts in an ontology are described according to the RDF model specification [7]. Semantic meta-data are stated as collections of interlinked RDF triples in a RDF graph.
- The RDF graph is partitioned into disjoint subgraphs describing resources according to the *same subject criterium*. Each set of triples in the graph sharing the same subject are considered to be describing the same resource.
- Triples for each resource can be manipulated by a process at run time as triple spaces, using classical triple space operations[6]: `read`, `in`, `out`. Additionally the `subscribe` operation is defined.
- Each resource's triple space can be identified by the URI in the subject of the triples stored in that triple space. A RESTful semantic service can be described as a process receiving HTTP requests through the URI of an associated triple space and manipulating the triple space with triple space operations, according to the semantics of HTTP methods.
- RESTful semantic web services and clients can be described in an uniform way as processes with attached triple spaces exchanging HTTP messages through channels named after URIs. As a result of these messages, triples are transferred between services' triple spaces and clients' triple spaces. Clients can obtain access to new services through the URIs of the triples retrieved. Communication between clients and semantic RESTful services is modeled using an extension to the Pi-Calculus for Mobile Processes [13].

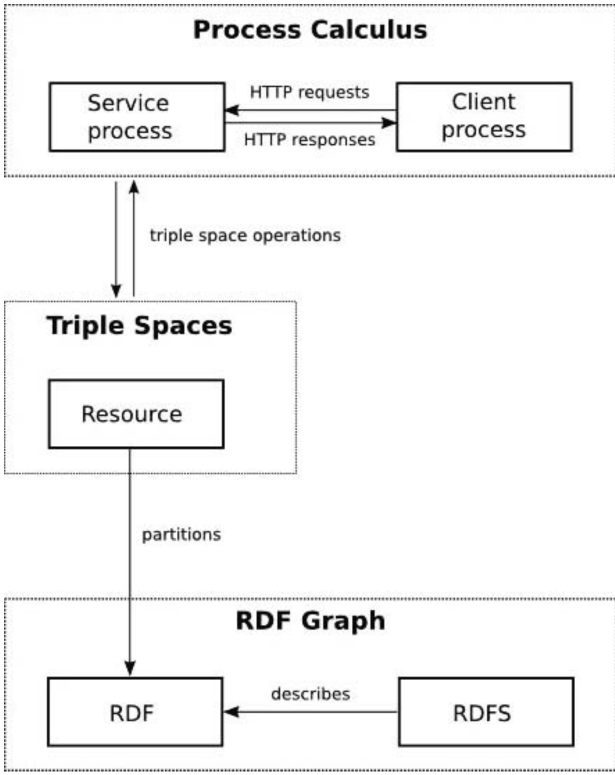


Fig. 1 Different conceptual levels used in the description of the RESTful semantic services model

Each level of the service model is concerned with a certain aspect of distributed computing. The RDF graph allows a powerful mechanism for the description of data involved in a distributed computation. The use of resources and triple spaces offers a finite set of simple yet powerful operations to manipulate the semantic data stored in the RDF graph. Additionally, using blocking operations like `in` or `subscribe` serve as implicit coordination and notification primitives for processes manipulating resources. Finally, a formal model like the proposed extension for the Pi-calculus [15] addresses the problems of the composition and coordination of RESTful semantic services and client processes.

3 A Process Calculus for RESTful Semantic Services

Interaction between RESTful semantic services and clients is defined in the service model as an extension to the polyadic Pi-Calculus [12]. The main constructs of the Pi-Calculus are present: processes, channels and messages. Those components represent concepts of the service model: clients and services processes, URIs and HTTP requests and responses.

In the original Pi-Calculus names were exchanged between processes through other names being used as channels. In our version triples containing URIs are exchange through channels named after URIs. The URIs retrieved from triples can be used to send messages to new processes. Besides triples, triple patterns can also be exchanged. An operation for matching patterns against triples is defined.

The main extension to the calculus is the introduction of the concept of triple space as shown in the following syntax specification:

P,Q,A,B	$::=$	REST services / agents
$\mid$	$w(u) < v >$	Writing of value v in the u triple space
$\mid$	$r(u) < p >$	Non blocking reading of values matching pattern p in the u triple space
$\mid$	$br(u) < p >$	Blocking reading of values matching pattern p in the u triple space
$\mid$	$d(u) < v >$	Deleting of value v in the u triple space
p,q	$::=$	Triple patterns
$u_1,u_2..u_n$	$::=$	Triplet space handlers
m	$::=$	HTTP messages
	$[GET,p,c]$	HTTP GET request with pattern p and response channel c
$\mid$	$[POST,p,c]$	HTTP POST request with pattern p and response channel c
$\mid$	$[PUT,v,c]$	HTTP PUT request with triplets of value T and response channel c
$\mid$	$[DELETE,(),c]$	HTTP DELETE request and response channel c

Each process in the calculus is not only defined by a calculus' expression but also for a set of triple spaces associated to the process. Operations for reading, writing and deleting triples from the associated triple spaces are defined as shown in the following collection of labeled transitions:

- T_OUT_1 : c,p when p has multiple matches in u, $\overline{c}[\_,r(u) < p >] \stackrel{r(u)<p>}{\longrightarrow} \overline{c}[\_,(v)]$
- T_OUT_2 : c,p when p has no match in u, $\overline{c}[\_,r(u) < p >] \stackrel{r(u)<p>}{\longrightarrow} \overline{c}[\_,()]$
- T_IN : v a set of triples, $TS < P > \vdash w < v > \stackrel{\tau}{\longrightarrow} |P| \cup \{v\} \vdash 0$
- T_DEL : v a set of triples $TS < P > \vdash d < v > \stackrel{\tau}{\longrightarrow} |P| - \{v\} \vdash 0$

4 Implementation of RESTful Semantic Web Services

As a test for the service model previously introduced, *Semantic Bespin*, an implementation of a web application consuming a composition of RESTful semantic web services has been built (source code and libraries can be found at `http://github.com/antoniogarrote/semantic_rest`) The Javascript interpreter running in Mozilla's Firefox web browser was chosen as the run time for the execution of the client part of the application. The web framework Ruby on Rails was chosen for building the RESTful semantic services that had to be consumed from the client. This was meant to be a very constrained environment where the viability of an implementation of the service model could be tested, but also a representative example of nowadays mainstream web development platforms.

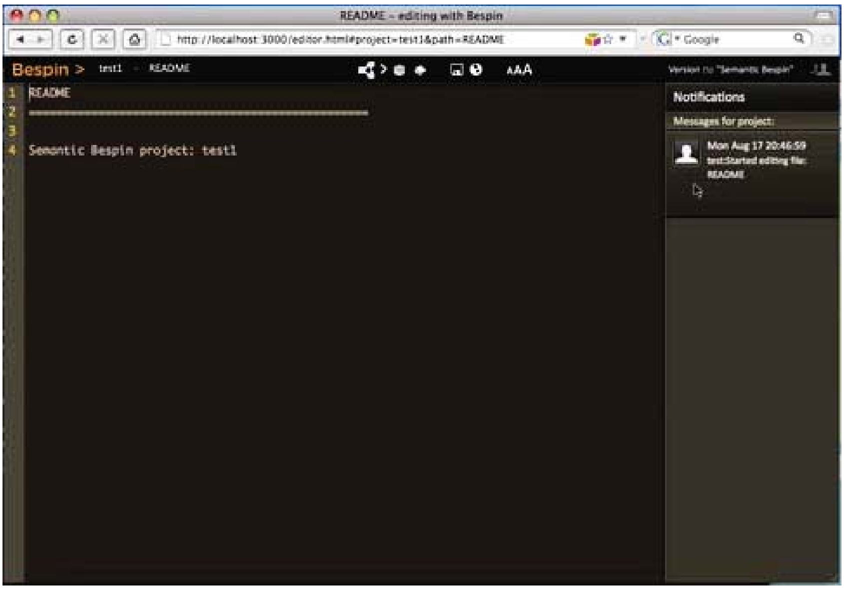


Fig. 2 Notifications are retrieved within Semantic Bespin consuming the Twitter RESTful semantic web service

Before building the actual implementation of the application, a concrete specification of the service model had to be chosen. This specification describes the abstract components of the web services model with an actual syntax for the messages client and service must exchange in order to consume the semantic meta data. It also defines how semantic meta-data must be transformed into a valid representation for the transport layer between client and server, as well as the operations for *lifting* and *lowering* [9] data between both representations.

The proposals MicroWSMO and hRESTS [8] were chosen. These standards make use of a subset of the WSMO ontology to describe REST architecture components. hRESTS was expanded adding new constructs to the hRESTS service description ontology:

- `parameters` are used to send additional information in a HTTP message that could modify the processing of that message by the client or server processing the message. These parameters are optional and can be safely ignored by a process not supporting the annotation.
- a set of these annotations were designed to add information regarding the processing of HTTP messages in a Javascript application.
- Javascript was used as an alternative to other mechanisms like XSLT as a mechanism for *lifting* semantic information from the transport syntax to a higher level encoding.

The application built consists of a code editor running in the web browser that integrates source version control and notifications between users.

These modifications were due to the fact that our client process was a Javascript application running inside the browser. We were forced to use techniques like *JSONP* to overcome limitations like the *same domain policy* imposed to HTTP requests for Javascript code being executed inside the browser. The use of JSON [3] as the basic exchange format was also required.

The extensions of hRESTS with parameters were designed to allow the specification of this kind of specific behaviour without breaking compatibility with the standard as shown in the following RDFS statements:

```
@prefix hr: <http://www.wsmo.org/ns/hrests#>.
@prefix hrjs: <http://semantic_rest.org/ns/hrests_js#>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.
@prefix wsl: <http://www.wsmo.org/ns/wsmo-lite#>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.

hr:InputParameter a rdfs:Class.
hr:hasInputParameter a rdf:Property;
  rdfs:domain wsl:Message;
  rdfs:range hr:InputParameter.
hr:parameterName a rdf:Property;
  rdfs:domain hr:InputParameter;
  rdfs:range xsd:String.

hrjs:JSONPCallback a hr:InputParameter.
```

The actual extensions defined for Javascript included the specification of a callback argument for JSONP. and the specification of HTTP methods not supported in web browsers like DELETE or PUT:

```
@prefix hr: <http://www.wsmo.org/ns/hrests#>.
@prefix hrjs: <http://semantic_rest.org/ns/hrests_js#>.
hrjs:JSONPCallback a hr:InputParameter.
hrjs:overloadedMethod a hr:InputParameter.
```

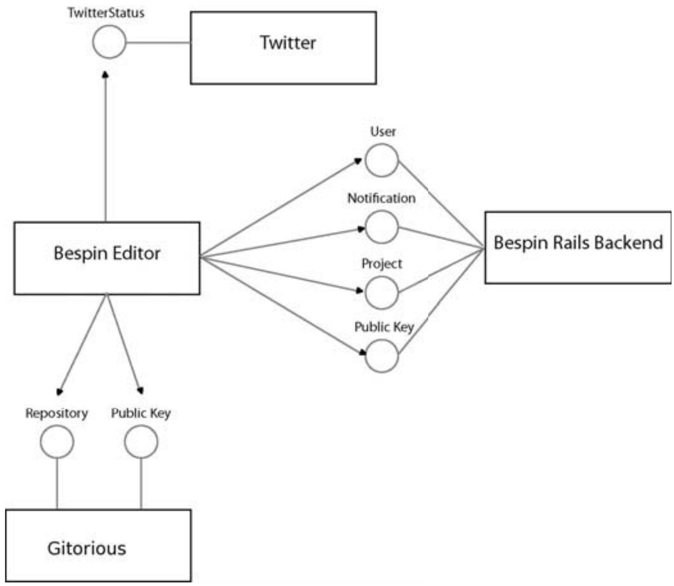


Fig. 3 Location of the services used in the application

Several services and applications were composed using RESTful semantic web services to build the application: Bespin Project from the Mozilla Foundation (`https://bespin.mozilla.com/`) as the Javascript client consuming the services, Gitorious (`http://gitorious.org/`) for building the services managing the source control repository and Twitter (`http://twitter.com/`) serving as the communication bus between users.

Some of these services were built from scratch, as the backend application for the Bespin Javascript client providing user and project resources. Others were built modifying the code of existent web applications to export part of the functionality as RESTful semantic resources as it was the case of Gitorious or Bespin. Finally Twitter service could only be wrapped with a description of the HTTP interface using hRESTS. Since Twitter API follows REST principles, the Javascript client was able to consume the service using only the description of the service provided.

5 Conclusions

In this article a semantic web services model inspired in the design principles of RESTful web architectures and triple space computing has been introduced. The main aim of this model is to enable the description of semantic web services architectures retaining the simplicity and power of REST architectures.

The proposed model builds on top of projects like the TripCom project [14], where triple space computing model is integrated within the WSMO framework for semantic web services. Our proposal extends the triple space computing model using a subset of WSMO, compatible with RESTful architectural principles.

Complex computation involving several RESTful semantic services and clients can be expressed using an extension of the Pi-Calculus with concepts from triple space computing and REST constructs. This theoretical model serves as the underlying formal basis for the systems built with our framework proposal.

Finally, the validity of this approach has been tested with the development of a sample web application, consisting of the aggregation of several RESTful semantic web services, as well as software libraries for Javascript and Ruby implementing the proposed model.

References

1. Akkiraju, R.: Web Service Semantics - WSDL-S (2005), http://www.w3.org/Submission/WSDL-S/
2. Battle, S.: Semantic Web Services Framework (SWSF) (2005), http://www.w3.org/Submission/2005/07/
3. Chinnici, R.: Request for Comment: 4627 (JSON) (2006), http://www.ietf.org/rfc/rfc4627.txt
4. Fensel, D.: Triple-space computing: Semantic Web Services based on persistent publication of information. In: IFIP International Conf. on Intelligence in Communication Systems, pp. 43–53 (2004)
5. Fielding, R.: Architectural Styles and the Design of Network-based Software Architectures. University of California, Irvine (2000)
6. Gelernter, D.: Generative Communication in Linda. ACM Transactions on Programming Languages and Systems 7, 80–112 (1985)
7. Hayes, P.: RDF Semantics (2004) , http://www.w3.org/TR/2004/REC-rdf-mt-20040210/
8. Kopecky, V.: hRESTS and MicroWSMO (2009), http://www.w3.org/TR/rdfa-syntax
9. Fensel, D.: Enabling Semantic Web Services: The Web Service Modeling Ontology. Springer, Heidelberg (2006)
10. Lausen, H.: Web Service Modeling Ontology (WSMO) (2005), http://www.w3.org/Submission/WSMO/
11. Martin, D.: OWL-S: Semantic Markup for Web Services (2004), http://www.w3.org/Submission/OWL-S/
12. Milner, R.: The Polyadic Pi-Calculus Logic and Algebra of Specification (1991)
13. Milner, R.: Communicating and Mobile Systems: the Pi-Calculus. Cambridge University Press, Cambridge (1999)
14. Nixon, L.: Triple Space as a Global Semantic Coordination Middleware. In: 7th IEEE International Workshop on Enabling Technologies: Infrastructures for Collaborative Enterprises (2008)
15. Lapadula, A.: A Calculus for Orchestration of Web Services. In: De Nicola, R. (ed.) ESOP 2007. LNCS, vol. 4421, pp. 33–47. Springer, Heidelberg (2007)

Enhancing the Services Integration Mechanism in the HoCa Multi-agent Architecture

J.A. Fraile, Jesús A. Román, and Belén Pérez-Lancho

Abstract. This work presents improvements in the communication protocol and service coordination mechanism of the HoCa architecture, a multi-agent based architecture designed to facilitate the development of pervasive systems that proposes a new model where multi-agent systems and service oriented architectures are integrated to facilitate compatible services. A coordinator agent proposes a new planning model, where the complex processes are modeled as external services. The agent acts as coordinators of Web services that implement the four stages of the case-based planning cycle. One of the main aims of the pervasive systems is to be able to adapt themselves in execution time to the changes in the number of resources available, the mobility of the users, variability in the needs of the users and failures of the system. Multiagent systems are suitable to resolve these issues due to their capabilities such as autonomy, reactivity, pro-activity, mobility, etc.

Keywords: Multi-Agent Systems, Services Oriented Architectures, Distributed Computing.

1 Introduction

The importance acquired by the dependency people sector has dramatically increased the need for new home care solutions [7]. Besides, the commitments that have been acquired to meet the needs of this sector, suggest that it is necessary to modernize the current systems. Multiagent systems [11], and intelligent devices-based architectures have been recently explored as supervisor systems for health care scenarios [1] [17] for elderly people and for Alzheimer patients [7]. These systems allow providing constant care in the daily life of dependent patients [4], predicting potentially dangerous situations and facilitating a cognitive and physical support for the dependent patient [2]. Taken into account these solutions, it is possible to think that multi-agent systems facilitate the design and

J.A. Fraile · Jesús A. Román · Belén Pérez-Lancho
Facultad de Ciencias
Universidad de Salamanca
Plaza de la Merced s/n, 37008, Salamanca, Spain
e-mail: jafraileni@upsa.es, {zjarg,lancho}@usal.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 303–310.
springerlink.com

development of pervasive environments [9] and improve the services currently available, incorporating new functionalities. Multi-agent systems add a high level of abstraction regarding to the traditional distributed computing solutions.

While the challenge of bringing smart environments to multi-agent scales is somewhat overwhelming, multi-agent settings also raise a number of opportunities for unique research [8]. The HoCa multi-Agent architecture [12] uses a series of components to offer a solution that includes all levels of service for various systems. It accomplishes this by incorporating intelligent agents, identification and localization technology, wireless networks and mobile devices. Additionally, it provides access mechanisms to multi-agent system services, through mobile devices, such as mobiles phones or PDA. The architecture integrates two types of agents, each of which behaves differently for specific tasks. The first group of agents is made up of deliberative BDI agents, which are in charge of the management and coordination of all system applications and services. The second group of agents is made up of reactive agents responsible for handling information and offer services in real time. The communications between agents within the platforms follows the FIPA ACL (Agent Communication Language) standard. The protocol for communication between agents and services is based on the SOAP standard.

The main objective of this paper is to improve the HoCa hybrid Multi-Agent architecture [12], for the control and supervision of pervasive environments [10]. The improvements focus on the communication protocol and the coordination mechanism for the services. In this way, the architecture provides innovative mechanisms to integrate multi-agent systems with service oriented architectures and intelligent interfaces to obtain context-aware information. The architecture incorporates technologies for automatic identification, location, alarms management and movement tracking. These technologies facilitate the monitoring and management of dependent patients at their home in a ubiquitous way. One of the main contributions of this paper is the use of coordinator deliberative BDI agents [7], specialized in the distribution of complex tasks.

The next section reviews the problem that motivates the majority of this research. Section three describes the communication protocol proposed for the HoCa architecture [12]. Section four presents new mechanism for the coordinator agent. Finally the conclusions and some preliminary results are presented.

2 Problem Description and Background

Context-aware systems provide mechanisms for developing applications that understand their context and are capable of adapting to possible changes. A context-aware application uses the context of its surroundings to modify its performance and better satisfy the needs of the user within that environment. The information is usually obtained by sensors. The current trend for displaying information to the system users, given the large number of small and portable devices, is the distribution of resources through a heterogeneous system of information networks. Web applications and services have been shown to be quite efficient [18] in processing information within this type of distributed system. Web applications are run in distributed environments and each part that makes up the program can be located in a different machine. Some of the web technologies that have had an important role over the last few years are multiagent systems and SOA (Service Oriented Architecture) architectures, which focus on the distribution of system service functionalities. This model provides a flexible distribution of resources and facilitates the inclusion of new functionalities within changing environments. In this respect, the multiagent systems have also already demonstrated their aptitude in dynamic changing environments [4] [10]. The advanced state of development for multiagent systems is making it necessary to develop new solutions for context-aware systems. It involves advanced systems that can be implemented within different contexts to improve the quality of life of its users. There have been recent studies on the use of multiagent systems [4] as monitoring systems in the medical care [2] patients who are sick or suffer from Alzheimer's [10]. These systems provide continual support in the daily lives of these individuals [11], predict potentially dangerous situations, and manage physical and cognitive support to the dependent person [4].

A multiagent system consists of intelligent entities that are called agents. An agent is a physical or abstract entity that can collect information through sensors, is able to assess such perceptions and make decisions through mechanisms of simple or complex reasoning, communicate with other agents to obtain information and act on the environment in that it operates through executing [18]. Multiagent systems can easily adapt their behavior to the changing context. These systems can use the latest technology in computer whenever necessary and to minimize network traffic, especially in wireless networks. A Multiagent system provides users with various wireless services that enhance the capabilities of mobile devices connected. These systems adapt in a transparent way the user interface to a specific platform, monitoring the movements of the user and communicate this information to agents on the platform. There are also a number of agents who are in charge of streamlining operations at the network and interact with other agents.

Multiagent systems are easily adaptable to pervasive environments [18]. The main function of a pervasive multiagent pervasive is to provide a framework for implementation of the agents who are part of it. Pervasive multiagent system consists of at least one channel of communication and a set of services that facilitate the interconnection between the agents that comprise it [17]. These services provided by the system are invisible to the user. This paper presents a communication protocol to integrate Web Services with multiagent systems and an intelligent agent for coordinating services in the HoCa architecture [12].

3 Communication Protocol to Integrate Web Services and Multiagent Systems

Communication protocol allows applications, services and sensors to be connected directly to the platform agents. The protocol presented in this work is open and independent of programming languages. It is based on the SOAP (Simple Object Access Protocol) standard and allows messages to be exchanged between applications and services as shown in Figure 1.

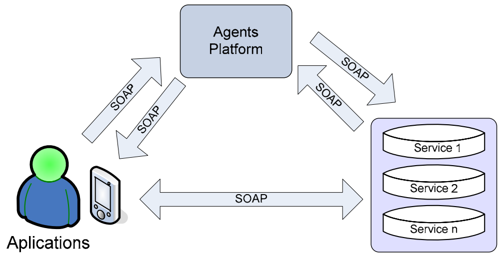


Fig. 1 Communication using SOAP messages in HoCa

SOAP is a standard protocol that defines how two objects in different processes can communicate through XML data exchange. For example, here are displayed as a HoCa user since the supervisor application, which can run on a PDA asks the agent CoAp (see Figure 4) the patient location in his home. The application requests the patient location with a SOAP message (see Alg. 1) and the CoAp agent when has the patient location gets to communicate with the application back in another SOAP message (see Alg. 2) the information requested.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
    <soap:Body>
      <getHCPatientLoc xmlns="http://hoca.example.com/lc">
        <patientId>6267581</patientId>
      </ getHCPatientLoc>
    </soap:Body>
</soap:Envelope>
```

Alg. 1 Request for patient location

In the algorithm 1 is seen as the application asks for the patient location through his identifier in the system. Both in the algorithm 1 as in the algorithm 2 are seen as the SOAP message structure consists of a header or envelope and content or message body.

```
<soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
    <soap:Body>
      <getHCPatientLocDetailsResponse
xmlns="http://hoca.example.com/lc">
        <getHCPatientLocDetailsResult>
          <patientName>Juan Nieto</patientName>
          <patientId>6267581</patientId>
          <description>Juan is in the dining room near the
TV</description>
          <location>835,264,176</location>
          <inHome>true</inHome>
        </getHCPatientLocDetailsResult>
      </getHCPatientLocDetailsResponse>
    </soap:Body>
 </soap:Envelope>
```

Alg. 2 Reply with the patient location

The algorithm 2 shows as the agent CoAp responds with more patient data such as name, house area description where he is and the patient location.

However, interaction with environmental sensors requires Real-time Transport Protocol (RTP)[14] which provides transport functions that are adapted for applications that need to transmit real-time data such as audio, video or simulation data, over multicast or uncast network services. The RTCP protocol is added to RTP, allowing a scalable form of data supervision. Both RTP and RTCP are designed to work independently from the transport and lower network services. They are in charge of transporting data with real-time characteristics, and of supervising the quality of service, managing the information for all the entities taking part in the current session.

The communications between agents within the platforms follows the FIPA ACL (Agent Communication Language) standard. This way, the applications can use the platform to communicate directly with the agents. The agent's messages structure are key-value row. These rows are written in an agent's communications language as FIPA ACL. The messages include the names of the sender and the receiver and may contain other messages recursively. Moreover defining protocols for high-level interaction between the agents, called talks and it is possible to define new primitives from a core of primitive by composition.

4 Agents with Advanced Services Coordination Abilities

Services represent the activities that the multi-agent architecture offers. Services are components of the architecture. The services are invoked on the applications through the agent's platform. They are the bulk of functionalities of the pervasive system at processing, delivery and acquire information level. Services are designed to be invoked locally or remotely. Services can be organized as local services, web services, or even as individual stand alone services. Services can make use of other services to provide the functionalities that users require. There is a flexible and scalable directory of services, so they can be invoked, modified, added, or eliminated dynamically and on demand. It is absolutely necessary that all services follow the communication protocol to interact with the rest of the architecture components.

The coordinator agent is the core of the system, since provides the ability for self-organization. The agents in the organization layer have the capacity to learn from the analysis carried out in previous procedures. They adopt the model of reasoning CBP, a specialization of case-based reasoning (CBR) [16]. CBP is the idea of planning as remembering [13]. In CBP, the solution proposed to solve a given problem is a plan, so this solution is generated taking into account the plans applied to solve similar problems in the past [16]. The problems and their corresponding plans are stored in a plans memory. A plan P is a tuple $<S,B,O,L>$, S is the set of plan actions, O is an ordering relation on S allowing to establish an order between the plan actions, B is a set that allows describing the bindings and forbidden bindings on the variables appearing in P, L is a set of casual links.

The CBP-BDI agents stem from the BDI model [5] and establish a correspondence between the elements from the BDI model and the CBP systems.

The BDI model adjusts to the system requirements since it is able to define a series of goals to achieve based on the information that has been registered with regards to the world. Fusing the CBP agents together with the BDI model and generating CBP-BDI agents makes it possible to formalize the available information, the definition of the goals and actions that are available for resolving the problem, and the procedure for resolving new problems by adopting the CBP reasoning cycle.

Based on this representation, the CBP-BDI coordinator agents combine the initial state of a case, the final state of a case with the goals of the agent, and the intentions with the actions that can be carried out in order to create plans that make it possible to reach the final state. The actions that need to be carried out are services, making a plan an ordered sequence of services. It is necessary to facilitate the inclusion of new services and the discovery of new plans based on existing plans. Services correspond to the actions that can be carried out and that determine the changes in the initial problem data. Each of the services is represented as a node in a graph. The presence of an arch that connects to a specific node implies the execution of a service associated with the end node. The plan described by a graph can be defined by a sequence (i.e.: $S_7 \, o \, S_5 \, o \, S_3 \, o \, S_1)(e_0)$. e_0 represents the original state that corresponds to Init, which represents the initial problem description e_0. Final represents the final state of the problem e^*.

CBP-BDI agents use the information contained in the cases in order to perform different types of analyses. As previously explained, an analysis assumes the construction of the graph that will determine the sequence of services to be performed.

5 Results and Conclusions

The architecture presented in this paper was used to develop a prototype in the home of a dependent person. It incorporates JavaCard technology to identify and control access, with an added value of RFID technology. The integration of these technologies makes the system capable of automatically sensing stimuli in the environment in execution time. As such, it is possible to customize the system performance, adjusting it to the characteristics and needs of the context for any given situation. Different studies related to context-aware systems, such as [15] [20], focus exclusively on gathering positional data on the user. Many of these signals work with a very wide positioning range, which makes it difficult to determine the exact position of the user. In contrast, the system presented in this paper determines the exact position of the user with a high level of accuracy. To do so, the system uses JavaCard and RFID microchip located on the users and in the sensors that detect these microchips in their context.

The architecture, in addition to locating the users in their context, try to improve the communication between patients and medical personnel in a hospital center by capturing context attributes such as weather, the state of the patient or role of the user. In addition to capturing information from various context attributes such as location, temperature and lighting, the architecture provide services proactively to the user within a Home Care environment.

Although there still remains much work to be done, the system prototype that we have developed improves home security for dependent persons by using supervision and alert devices. It also provides additional services that react automatically in emergency situations. As a result, we have create a context-aware system that facilitates the development of intelligent distributed systems and renders services to dependent persons in their home by automating certain supervision tasks and improving quality of life for these individuals. The use of a multi-agent system, web services, RFID technology, JavaCard and mobile devices provides a high level of interaction between care-givers and patients. Additionally, the correct use of mobile devices facilitates social interactions and knowledge transfer. Our future work will focus on obtaining a model to define the context, improving the proposed prototype when tested with different types of patients.

Adopting a multi-agent software architecture for smart environments opens up a number of new research directions. One challenge for multi-agent smart environment software architectures is to define lightweight, simple, and scalable methods for communicating between the agents [8]. HoCa [12] provides an integral Ambient Intelligence-based solution for Home Care. The architecture facilitates the development of distributed environments using RFID, JavaCard and mobile devices, providing a high level of interaction with the users and patients, which is an essential factor in building pervasive environments. Moreover, the incorporation of CBR-BDI agents facilitates automatic decision making, as well as great capacities for learning and adaptation. However, the system needs to be tuned and evaluated in different real environments. That is our next challenge.

Acknowledgements. This work has been supported by the Spanish Ministerio de Ciencia e Innovación project PET2008_0036 projects.

References

1. Angulo, C., Tellez, R.: Distributed Intelligence for smart home appliances. Tendencias de la minería de datos en España. Red Española de Minería de Datos. Barcelona, España (2004)
2. Ardissono, L., Petrone, G., Segnan, M.: A conversational approach to the interaction with Web Services. Computational Intelligence 20, 693–709 (2004)
3. Augusto, J.C., McCullagh, P.: Ambient Intelligence: Concepts and Applications. Invited Paper by the International Journal on Computer Science and Information Systems 4(1), 1–28 (2007)
4. Bahadori, S., Cesta, A., Grisetti, G., Iocchi, L., Leone1, R., Nardi, D., Oddi, A., Pecora, F., Rasconi, R.: RoboCare: Pervasive Intelligence for the Domestic Care of the Elderly. AI*IA Magazine Special Issue (January 2003)
5. Bratman, M.: Intention, Plans and Practical Reason. Harvard U.P., Cambridge (1987)
6. Carrascosa, C., Bajo, J., Julian, V., Corchado, J.M., Botti, V.: Hybrid multi-agent architecture as a real-time problem-solving model. Expert Systems With Applications 34(1), 2–17
7. Chan, M., Campo, E., Esteve, D.: Assessment of activity of elderly people using a home monitoring system. International Journal of Rehabilitation Research 28(1), 69–76 (2005)

8. Cook, D.J.: Multi-agent smart environments. Journal of Ambient Intelligence and Smart Environments 1, 51–55 (2009)
9. Corchado, J.M., Laza, R.: Constructing Deliberative Agents with Case-based Reasoning Technology. International Journal of Intelligent Systems 18, 1227–1241 (2003)
10. Corchado, J.M., Bajo, J., Abraham, A.: GERAmI: Improving the delivery of health care. IEEE Intelligent Systems. Special Issue on Ambient Intelligence (March/April 2008)
11. Corchado, J.M., Bajo, J., de Paz, Y., Tapia, D.: Intelligent Environment for Monitoring Alzheimer Patients, Agent Technology for Health Care. Decision Support Systems 34(2), 382–396 (2008)
12. Fraile, J.A., Bajo, J., Corchado, J.M.: Multi-Agent Architecture for Dependent Environments. Providing Solutions for Home Care. Inteligencia Artificial 42, 36–45 (2009)
13. Glez-Bedia, M., Corchado, J.: A planning strategy based on variational calculus for deliberative agents. Computing and Information Systems Journal 10(1), 2–14 (2002)
14. Jacobsen, V., Fredrick, R., Casner, S., Schulzrinne, H.: RTP: A transport Protocol for Real-Time Applications. RFC 1889. Lawrence Berkeley National Laboratory, Xerox PARC, Precept Software Inc., GMD Fokus (January 1996), `http://www.connect.org.uk./teckwatch/cgi-bin/rfcshow?1889` (Accessed October 16, 1996)
15. Kerer, C., Dustdar, S., Jazayeri, M., Gomes, D., Szego, A., Caja, J.A.B.: Presence-aware infrastructure using web services and RFID technologies. In: Proceedings of the 2nd European Workshop on Object Orientation and Web Services, Oslo, Norway (2004)
16. Kolodner, J.: Case-Based Reasoning. Morgan Kaufmann, San Francisco (1993)
17. Moreno, A., Valls, A., Isern, D., Sanchez, D.: Applying Agent Technology to Healthcare: The GruSMA Experience. IEEE Intelligent Systems, 63–67 (2006)
18. Oren, E., Haller, A., Mesnage, C., Hauswirth, M., Heitmann, B., Decker, S.: A Flexible Integration Framework for Semantic Web 2.0 Applications. IEEE Software 24(5), 64–71 (2007)
19. Prentza, A., Maglavera, S., Stalidis, G., Sakka, E., Lekka, I., Angelidis, P.A., Leondaridis, L., Maglaveras, N., Koutsouris, D.: Cost-effective health services for interactive lifestyle management: the PANACEIA-iTV and the e-Vital concepts. Journal of Telecommunications and Information Technology, 49–58 (2005)
20. Priyantha, N.B., Chakraborty, A., Balakrishnan, H.: The cricket location-support system. In: Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, pp. 32–43. ACM Press, New York (2000)

Integrating Deep-Web Information Sources*

Iñaki Fernández de Viana, Inma Hernandez, Patricia Jiménez,
Carlos R. Rivero, and Hassan A. Sleiman

Abstract. Deep-web information sources are difficult to integrate into automated business processes if they only provide a search form. A wrapping agent is a piece of software that allows a developer to query such information sources without worrying about the details of interacting with such forms. Our goal is to help software engineers construct wrapping agents that interpret queries written in high-level structured languages. We think that this shall definitely help reduce integration costs because this shall relieve developers from the burden of transforming their queries into low-level interactions in an ad-hoc manner. In this paper, we report on our reference framework, delve into the related work, and highlight current research challenges. This is intended to help guide future research efforts in this area.

Keywords: Information, web, integration.

1 Introduction

Our work focuses on deep-web information sources that provide advanced search forms to build search constraints using a number of search fields, e.g., title, author, or price [23]. Our goal is to provide the technology a developer requires to develop agents that can integrate these sources into typical business applications. Such integration is usually addressed by means of wrappers, which are software agents that provide an API that abstracts developers from the details required to simulate a human interacting with a search form.

Iñaki Fernández de Viana · Patricia Jiménez
University of Huelva
e-mail: {i.fviana,patricia.jimenez}@dti.uhu.es

Inma Hernandez · Carlos R. Rivero · Hassan A. Sleiman
University of Sevilla
e-mail: {inmahernandez,carlosrivero,hassansleiman}@us.es

* Supported by the European Commission (FEDER), the Spanish and the Andalusian R&D&I programmes (grants TIN2007-64119, P07-TIC-2602, P08-TIC-4100, and TIN2008-04718-E).

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 311–320.

Note that typical wrapping agents do not relieve developers from the burden of transforming their queries into a number of actions on the search form. Virtual integration techniques, a.k.a. metasearch, provide a means to increase the abstraction level since they allow to create a unified search form that abstracts away from the details of several related actual forms in a given domain, e.g., flights, hotels, or bibliography [6]. These techniques increase the abstraction level, since developers only need to map their queries onto a unified search form, i.e., they are relieved from the burden of simulating the interaction with actual search forms.

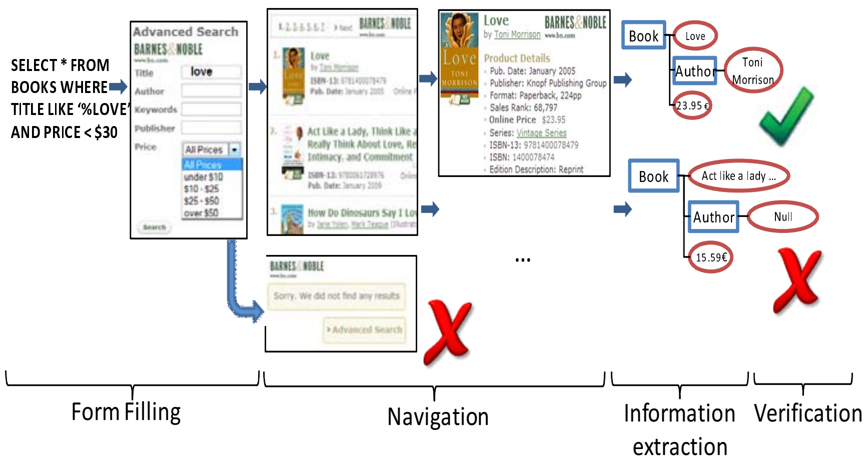


Fig. 1 An overall image of our proposal

Our hypothesis is that the effort might be reduced further if wrapping agents were able to understand higher-level queries written in languages such as SQL or SPARQL. As a motivating example, consider the bookstore scenario in Figure 1, in which a user searches for books that contain "love" in the title and cost less than $30. Answering this query goes through the following: 1) form filling, i.e., the query is analysed to find out how to fill in the search form appropriately; 2) navigation, i.e., the search button is pressed and the resulting page is navigated until pages about books are found; 3) information extraction (IE), i.e., once book pages are retrieved, the information of interest is extracted and structured according to a given ontology; 4) verification, i.e., to check the retrieved information for errors.

In this paper, we report on a proposal that allows to build wrapping agents with the above capabilities. Our goal is not to dive into details, but to provide an overall picture in which the emphasis is on making it explicit what the related work is and what the research challenges are. We expect these analysis to guide future efforts regarding building wrapping agents. In Sections 2–5, we delve into the details concerning each of the previous phases; in Section 6, we present our conclusions.

2 Form Filling

In this phase, the wrapping agent takes a high-level structured query over a deep-web information source as input and it has to translate the query into a number of search forms filled in with the suitable values. The first problem to be addressed is to use a semantic model of the search form, which is not machine-processable.

Existing approaches model a search form using several semantic levels [11]. One level deals with the query capabilities of the search form, i.e., what type of queries are issued to the information source through its search form. The query capabilities of a search form are modelled by parameterised views over the source [26, 27]. Some approaches deal with the automatic extraction of the search form query capabilities. Shu et al. [31] extract them by issuing predefined queries that help detect mandatory fields. Zhang et al. [36] extract hidden database attributes, operators which are applied to these attributes, and their ranges. Attributes are used in conjunctive queries because it is enough to capture a wide range of query capabilities.

When a query is posed over a deep-web information source, it has to be answered using only the views offered by the source. This problem is addressed by techniques for answering queries using views [10], which are based on selecting a number of views to answer a query. Another issue is known as heterogeneities in the predicate level [37], which happens when a query and the source may use different predicates for the same attribute, e.g., the query has a predicate: 'bookPrice $<$ \$15'; and the source accepts: 'bookPrice $<$ \$10' or 'bookPrice between \$10 and \$25' (cf. Figure 1). In this case, it is needed to fill two search forms in, one with 'bookPrice $<$ \$10' and another with 'bookPrice between \$10 and \$25', but a filter is needed in the second one to remove books whose price exceed \$15. To solve this problem, Zhang et al. [37] use a predicate mapping based on data types: depending on the data type of the attribute, there are a number of handlers that solve the heterogeneity problem.

An important aspect in the process of answering queries using views is to analyse the query feasibility, i.e., to study whether a query can be issued without executing it using the search form query capabilities. This analysis avoids a trial and error process in which the user writes a query and executes it until a suitable query is obtained. Petropoulos et al. [27] present a user interface for building SQL queries over a set of parameterised views that warns when a query is not feasible. Pan et al. [26] report on a generic framework for representing query capabilities that analyses the feasibility of SQL queries over deep-web information sources. An implementation of this framework and a recap on its main drawbacks is presented in [30].

After applying the techniques for answering queries using views and obtaining a number of ground views, each ground view can be seen as a search form that is filled in. The next step consists of actually filling each search form in with the values specified in each view. To perform this task, the next semantic level of existing search form models deals with the relations between the search form fields and the attributes [11], e.g., if the source contains an attribute 'publicationDate', this date can correspond to three fields in the search form: the day, the month and the year; these three fields are semantically related with the attribute 'publicationDate'.

3 Navigator

As we saw in the introduction, (cf. Figure 1), the navigation agent is responsible for reaching the pages relevant to the query, discarding or processing any other intermediate or error pages that may appear in the navigation sequence. For example, a no-results page when the information source does not have any information relevant to the query, an error page when a web application server raises and exception, or a disambiguation page in which the user is asked to clarify the query.

Traditional exhaustive crawlers [29] take a blind approach at navigation, following every link in each page. This is useful for certain tasks, like indexing pages for a search engine, but for virtual integration purposes it has an important drawback: usually web pages contain a large number of links, some leading to relevant information, but most having other purposes, like advertising or internal site navigation.

Virtual integration systems retrieve information online, hence system response time should be reasonably fast. Traditional crawlers are not suitable for virtual integration tasks, since crawling every single site means retrieving, analysing and classifying thousands of pages, most of them useless for this task. This results in an increment in cost and time that should be avoided.

As opposed to blind navigation, other approaches include some criteria to decide which are the links that must be visited, therefore reducing the number of irrelevant visited links. Relevancy criteria can be handcrafted by the user or automatically decided by the navigator, with the support of some reasoning process, usually in the form of a classifier [33]. The latter is the focus of our research, as we are interested in following only relevant links and retrieving only relevant pages. This kind of navigation is more efficient and is less costly than traditional crawlers.

Next, we briefly describe and analyse the existing proposals in the Navigation area. All of them define a navigation sequence, but we can distinguish between supervised (recorders) and non-supervised proposals (automated navigators), although we must note that, to the best of our knowledge, there is not a completely non-supervised proposal.

Recorders [1, 2] are the most supervised proposals. They present a simple user interface to record a user's navigation steps through an information source and store them in a file, in order to replay them automatically in future. These proposals interact directly with the web browser interface, so they do not deal with issues like scripts, posting forms, required authentication, or sites that keep session information. They depend completely on the user's knowledge, who is responsible for defining the navigation sequences, providing the values to fill in the forms and redefining the sequences whenever the target web site changes. Recording navigation steps makes navigation inflexible, and error-prone. However, it produces more efficient navigation patterns than traditional crawlers.

User-Defined Navigation [4, 9, 25] is less supervised than recorders, but still the system learns the in a supervised way, i.e., the user demonstrates how to obtain the relevant information, and the system is able to generalise this knowledge.

Automated navigators [3, 19, 21, 33] extract navigation sequences automatically from information sources, instead of learning them from the user. In order to accomplish this, the user has to provide some examples, but in this case the system needs less information than in the former proposals, like one or two examples of the relevant pages to be reached, or the set of keywords identifying them, inter alia.

Our focus is on following the least supervised paradigm. Therefore, our navigator agent crawls every information source. In every step (web page), the agent classifies it, and according to this class, a specifical set of actions is performed, in order to reach another page (e.g., clicking on a link or submitting a form). Once the agent has reached a relevant page, it is stored for latter processing. This is a lightly supervised system because the user only provides a reduced number of examples to train the classifier and to teach the navigator which links to follow and which ones to ignore.

4 Information Extractor

Results provided by the navigation module contain web pages in which data of interest is formatted in HTML. This format is easily interpreted by humans but agents face a difficulty in finding and extracting data of interest from this type of documents. The key is to use information extractors, which are algorithms that extract data of interest from the Web simplifying and reducing costs of the extraction task.

Despite the variety of proposals about IE algorithms, e.g. [8, 13, 12, 16], none of these solutions is universally applicable, and in an integration process, more than one proposal might be used. Besides, effectiveness and efficiency results are rarely comparable since they were obtained over different data sets.

Proposals are usually built using and providing different interfaces and technology, hence side-by-side comparisons are a tedious task since all these proposals should be implemented using same technology and tested over same data sets. Existing studies and surveys, such as [5, 14, 18], use taxonomies and attributes that don't provide any new information about extraction algorithms.

Our solution is an IE framework. IntegraWeb provides and uses a framework where all existing information extractors and perhaps the majority of new IE proposals can work side-by-side and whose results can be compared for each case of use. Here is a brief description of some of the framework components:

- Preprocessor: A large number of IE algorithms preprocess input documents before deploying them for learning or even for extraction. Cleaning the DOM tree, part of speech tagging or even separating DOM trees and removing unnecessary trees using [22] are typical preprocessing techniques.
- TrainingSet: Algorithms that learn rules for extraction need a training set that contains a set of samples marked by a user which are then used to infer extraction rules. Also, a configurable tokeniser should be used to tokenise the input samples.
- Learner: Algorithms that infer extraction rules use a learner. The learner can use a set of preprocessors, and some other utilities such as string or tree alignment classes. Besides, predefined algorithms are provided where user can define his algorithm's policy by implementing some template methods. For example, Branch and Bound algorithm is predefined and user should only define its main methods.

- WorkingSet: It contains an information extractor, input documents and generates a ResultSet. The information extractor is executed over input documents to extract the data of interest and structure it by means of attributes, slots and records.

We believe that an IE framework is the best solution for IntegraWeb since more than one information extractor could be needed depending on the web sites we are integrating. Our framework provides necessary components for developing existing and new IE algorithms. Developing an information extractor using our framework reduces costs since many reusable components can be used and there is no need to start from blank every time a new IE algorithm is developed.

Apart from the framework, we are working on a new algorithm to induce information extractors that is based on FOIL [28], which is a technique to induce first-order rules. FOIL is a climbing algorithm that starts from an empty rule that includes just a clause with a predicate that characterises a piece of information to be extracted and uses a heuristic that is based on a modified version of the information gain criterion to guide the search of clauses that can be used to complete the rule. A typical rule looks as follows: title(X) :- bold(X), link(X), next(X, A), author(A); intuitively, it means that a title in the web page is a piece of text annotated into the html document as bold and hyperlink; and that it is followed by another piece of text that has been previously identified as an author, whose rule also should be defined.

Unfortunately, the search space FOIL explores grow exponentially on the number of predicates available to characterise a piece of information. This motivated us to work on a number of optimisations and heuristics that may help to reduce this space. For instance, to define the semantics of predicates to detect inconsistencies amongst predicates in a rule or to reduce the number of new predicates that are going to be created and evaluated. Further, we use heuristics to define the order of in which predicates are explored by prioritising those that determine the left and right part of the target information (like next and previous) proposed in the extractors of Kushmerick [16] or using feature selection techniques to identify the most useful predicates depending on the studied domain. Therefore we translate the rules obtained on regular expressions that are easily undertandable by a machine.

5 Verifier

When information extractors are composed of extraction rules that rely on HTML land marks, they can only extract information from the same information source where the training was performed. Therefore, if the source rendering changes then the returned data could be incorrect. Unless the information generated by wrapping agents is verified in an automatic way, these data can go unnoticed for the applications using them.

On our analysis of the current literature, we have built a general verification framework composed by the following: Reaper, Assembler and Verification model Builder. The Reaper and the Assembler collaborate to generate a collection of valid result sets. This collection is used to infer a verification model that characterises the features of the correct data. Furthermore, we deal with the possibility of introducing perturbations to generate incorrect result sets: models that are built only on correct data leads to a overgeneralisation problem [34].

Regarding the Verification model Builder, a verification model is a characterisation of a training set that builds on the analysis of a number of features. Features are quantifiable characteristics and their values can be used as a form of evidence to decide if a result is valid or not. Features can be classified along two orthogonal axes: whether they are numeric or categorical, and whether they are applicable to slots or result sets. Numeric features transform slots or result sets into real numbers. The literature [35] reports on many numeric features, so we have grouped into several categories that range from counting the number of slots of a given class to counting attributes of a given class that match a given starting or ending pattern. Categorical features [24, 7] range from patterns that describe the structure of a record to constraints on the values of some attributes.

When models are constructed, a function has to be inferred from the training set, which should be constructed such that for a given feature vector x, an estimate of its quality is obtained, i.e., if this vector is similar to the rest of the training set.

In [20] the training set is characterised by a vector in which every feature is associated with its average value in the training set. In [15, 17] features are modelled as if they were random variables whose Gaussian distributions can be inferred from the training set; thus, to profile the value of a feature on an unverified result set, one can compute the probability that the corresponding random variable takes this value. The technique presented in [24] models every feature as if it was a random variable with a Gaussian distribution, but the profiles are calculated as the probability that a feature might have another value with a higher probability.

There are chances that alarms report false positives. Our approach in this cases is to use sanity checks, e.g., they use the Web as an information source to check if the data that has triggered the alarm is correct but infrequent.

6 Conclusion and Future Work

In this paper, we present a reference framework to build wrapping agents, which are pieces of software that query deep-web information sources. We focus on helping software engineers to construct wrapping agents that accept high-level structured queries. Our reference framework is composed of four phases and we highlight current research challenges in each phase.

In the form filling phase, the research challenges are:

1. Semantic models of the search forms are a cornerstone for the form filling task. In the bibliography, there are some proposals that devise complex semantic models of search forms but they fail on representing search forms using web technologies such as Javascript or AJAX. These technologies make search forms more interactive, i.e., a field can be hidden depending on the value of other field.
2. In the bibliography, a search form is seen as a number of parameterised views over the source, and a query over a deep-web information source is answered by applying the techniques of answering queries using views. These techniques are

mainly based on the relational or XML model but they have to be adapted to the Semantic Web model, where the concept of view is not so well-known.

Regarding the navigation phase, the main challenges are:

1. Response pages have to be classified into the different roles that they play. There are many proposals dealing with the web page classification problems, hence the problem comes down to choosing between one of them.
2. Links leading to relevant pages have to be identified before clicking on them, to avoid visiting useless pages.
3. The navigator should interact with the user as little as possible. Therefore, learning is unsupervised, or at least, very little supervised.
4. Instead of building an ad-hoc navigation model for every site, our focus is on developing a general model that can adapt to most sites just by tuning some parameters, and preferably change-resilient.

Besides mentioned challenges, here are some research challenges concerning IE:

1. The construction of an universal framework where earlier and new proposals can be integrated.
2. A survey that classifies information extractors to compare effectiveness and that uses comparable results over the same data set to compare efficiency.
3. Optimisation and improvement of existing IE algorithms.
4. An universally applicable effective and efficient information extractor.

Last, but not least, the verifying phase presents the following research challenges:

1. The verification modelling techniques described assume that the data sets returned by the reaping plan are homogeneous. To work with truly homogeneous data sets we propose to analyse the training set data and to obtain a series of new data sets, which will be homogeneous. It is interesting to study also the candidate features set before creating the verification model to reduce its size.
2. For the verifier training to be adequate, it is advisable that the training set has both valid and invalid examples. This is a problem as the training set has only valid examples.
3. We cannot assume all features follow a normal distribution and hence we must find techniques that allow modelling without any assumptions of its distribution.
4. The wrapper verification problem is closely related to the novelty recognition problem [32] if it is rephrased in terms of feature vectors and their similarities.

References

1. Anupam, V., et al.: Automating web navigation with the webvcr. Computer Networks 33(1-6) (2000)
2. Baumgartner, R., et al.: Deep web navigation in web data extraction. In: CIMCA/IAWTIC (2005)
3. Blanco, L., et al.: Efficiently locating collections of web pages to wrap. In: WEBIST (2005)

4. Blythe, J., et al.: Information integration for the masses. J. UCS 14(11) (2008)
5. Chang, C.-H., et al.: A survey of web information extraction systems. IEEE Trans. Knowl. Data Eng. 18(10) (2006)
6. Chang, K.C.-C., et al.: Toward large scale integration: Building a metaquerier over databases on the web. In: CIDR (2005)
7. Chidlovskii, B., et al.: Documentum eci self-repairing wrappers: performance analysis. In: SIGMOD Conference (2006)
8. Crescenzi, V., et al.: Roadrunner: Towards automatic data extraction from large web sites (2001)
9. Davulcu, H., et al.: A layered architecture for querying dynamic web content. In: SIGMOD Conference (1999)
10. Halevy, A.Y., et al.: Answering queries using views: A survey. VLDB J. 10(4) (2001)
11. He, H., et al.: Towards deeper understanding of the search interfaces of the deep web. World Wide Web (2007)
12. Hogue, A., Karger, D.R.: Thresher: automating the unwrapping of semantic content from the world wide web. In: WWW (2005)
13. Hsu, C.-N., Dung, M.-T.: Generating finite-state transducers for semi-structured data extraction from the web. Inf. Syst. 23(8) (1998)
14. Jung, K., et al.: Text information extraction in images and video: a survey. Pattern Recognition 37(5) (2004)
15. Kushmerick, N., et al.: Regression testing for wrapper maintenance. In: AAAI/IAAI (1999)
16. Kushmerick, N., et al.: Wrapper induction: Efficiency and expressiveness. Artif. Intell. 118(1-2) (2000)
17. Kushmerick, N., et al.: Wrapper verification. World Wide Web 3(2) (2000)
18. Laender, A.H.F., et al.: A brief survey of web data extraction tools. SIGMOD Record 31(2) (2002)
19. Lage, J.P., et al.: Automatic generation of agents for collecting hidden web pages for data extraction. Data Knowl. Eng. 49(2) (2004)
20. Lerman, K., et al.: Wrapper maintenance: A machine learning approach. Journal of Artificial Intelligence Research 18 (2003)
21. Liddle, S.W., et al.: Extracting data behind web forms. In: Spaccapietra, S., March, S.T., Kambayashi, Y. (eds.) ER 2002. LNCS, vol. 2503. Springer, Heidelberg (2002)
22. Liu, B., et al.: Mining web pages for data records. IEEE Intelligent Systems 19(6) (2004)
23. Madhavan, J., et al.: Harnessing the deep web: Present and future. In: CIDR (2009)
24. McCann, R., et al.: Mapping maintenance for data integration systems. In: VLDB (2005)
25. Montoto, P., et al.: A workflow language for web automation. J. UCS 14(11) (2008)
26. Pan, A., et al.: A model for advanced query capability description in mediator systems. In: ICEIS (2002)
27. Petropoulos, M., et al.: Exporting and interactively querying web service-accessed sources: The clide system. ACM Trans. Database Syst. 32(4) (2007)
28. Quinlan, J.R., et al.: Learning first-order definitions of functions. J. Artif. Intell. Res. (JAIR) 5 (1996)
29. Raghavan, S., Garcia-Molina, H.: Crawling the hidden web. In: VLDB (2001)
30. Rivero, C., et al.: From queries to search forms: an implementation. IJCAT 33(4) (2008)
31. Shu, L., et al.: Querying capability modeling and construction of deep web sources. In: Benatallah, B., Casati, F., Georgakopoulos, D., Bartolini, C., Sadiq, W., Godart, C. (eds.) WISE 2007. LNCS, vol. 4831, pp. 13–25. Springer, Heidelberg (2007)
32. Tax, D.M.J., et al.: One-class classification, concept learning in the absence of counter example. PhD thesis, Delft University of Technology (2001)

33. Vidal, M.L.A., et al.: Structure-based crawling in the hidden web. J. UCS 14(11) (2008)
34. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations (1999)
35. Wong, T.-L., Lam, W.: Adapting web information extraction knowledge via mining site-invariant and site-dependent features. ACM Trans. Internet Techn. 7(1) (2007)
36. Zhang, Z., et al.: Understanding web query interfaces: Best-effort parsing with hidden syntax. In: SIGMOD Conference (2004)
37. Zhang, Z., et al.: Light-weight domain-based form assistant: Querying web databases on the fly. In: VLDB (2005)

Ontological Trading in a Multi-agent System

J. Criado, N. Padilla, L. Iribarne, J.A. Asensio, and F. Muñoz

Abstract. In a more *open world*, the Web-based Information Systems (WIS) must be flexible and ready to be adaptable, extendable, accessible and operable by different people (or groups) who are in different places with different types of information (*convergent systems*). The convergence of systems is possible due to: (a) certain capacity for autonomy (*software agents*), (b) ability to use a common vocabulary for all convergent systems (*ontologies*); and (c) capacity for mediation between subsystems (*traders*). Moderns IS use ontologies as part of the design and implementation of their architecture. Traders are an essential part of these IS, where ontologies are required for them (traders) to facilitate integration. This paper describes the SOLERES trading service, a solution to carry out the mediation function within a multi-agent *environmental* information system. The data handled by the trader are defined by ontologies, as are the semantic description and content of the messages by which the trader communicates with the rest of the agents. An example to explain the functionality of the ontological trading agent is also introduced.

1 Introduction

In a more *open world*, Web-based Information Systems (WIS) must be flexible and ready to be adaptable, extendable, accessible and operable by different people or groups of people who are in different places with different types of information, facilitating access to the information, decision-making, workgroup, etc. (*convergent systems*). The convergence between systems is possible due to three basic parameters: (a) certain capacity for autonomy (**software agents**), (b) ability to use a common vocabulary for all convergent systems (**ontologies**); and (c) capacity for mediation between subsystems, necessary to allow, coordinate, translate and keep this common

J. Criado · N. Padilla · L. Iribarne · J.A. Asensio · F. Muñoz
Applied Computing Group, University of Almeria, Ctra Sacramento s/n, 04120 Almeria, Spain
e-mail: {crj282,npadilla,liribarne,jacortes,francijo}@ual.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 321–329.
springerlink.com

vocabulary (**traders**). IS often require the inclusion of a mediation component (i.e., *trader*). This is because many IS are found in distributed environments with local databases that need to be intercommunicated for an overall response.

At the present time there is a diversity of WIS with trading services implemented through multiagent systems. The ODP (Open Distributed Processing) trading function standard [1] defines five functional interfaces. The *Lookup* interface provides operations for importing trading service offers. The *Register* interface provides actions for managing the set of offers stored by the trader. The *Proxy* interface makes actions analogous to those offered by the Register interface, but modifies the set of proxy offers. The *Link* interface provides operations for creating and modifying links to other trader objects. Finally, the *Admin* interface offers actions that modify the trader operating policy attributes. There are a large number of studies in which the *trading service* follows the ODP specification. For instance, in [2] presents a trading service called DOKTrader, which acts on a federated database system called the Distributed Object Kernel (DOK). Another example is found in [3]. This study concentrates on the creation of a framework for developing distributed applications for a Common Open Service Market (COSM), making use of a Service Interface Description Language (SIDL) to describe the services manipulated by the trader. These approaches of the trading OPD implementations have several shortcomings like component interactions, object communications or language description, which have been improved using ontologies.

In recent years, the use of *ontologies* in trading services in WIS has spread, especially in web information services. Ontologies are being used to describe the services offered as well as communication primitives employed by system components. In [4] authors present the design of a market managed by ontologies. Within this system, an ontological communications language is used to represent queries, offers and agreements. Furthermore, in [5], ontologies are used to describe information shared by different system components. To achieve greater operability and autonomous, reactive component action, many systems have chosen to encapsulate the trader object within a software agent. [6] describes the MinneTAC agent, a trading agent developed to participate in the Trading Agent Competition (TAC). Through the description of this agent, implementation of a trader as a software agent is shown to maximize benefits from scenarios that require cooperation and negotiation between the trader and the rest of the system components, as well as systems that require communication among various trading objects, making use of ontologies to represent information shared by the agents, whether to describe data and the relationships among variables, as is the case in [7], or defining communication primitives and interaction among agents [8].

This paper describes the SOLERES trading service, which was designed to carry out the mediation function within a multiagent *environmental* information system. The data handled by the trader are defined by ontologies, as are the semantic description and content of the messages by which the trader communicates with the rest of the agents.

The rest of the paper is organized in the following manner. Section 2 describes the trading service implemented in our SOLERES multiagent system. Section 3 describes how the trading agent uses the ontologies. Finally, in Section 4, a scenario is given as an example in which one of the trader interface operations is executed. The paper ends with a series of conclusions and some proposals for further work.

2 Trading-Based Multiagent System

The trading service implemented in the SOLERES system is based on ODP trading function specifications (Table 1), in particular: instead of handling service offers, our trader manages EID documents that contain Environmental Information meta-metaData. The prototype developed currently implements a standalone trading service (with the Lookup, Admin and Register interfaces). As communication with other traders is unnecessary, the Link and Proxy interfaces have not been implemented. Our trading service has been implemented with software agents. Figure 1 shows the metamodel for this agent.

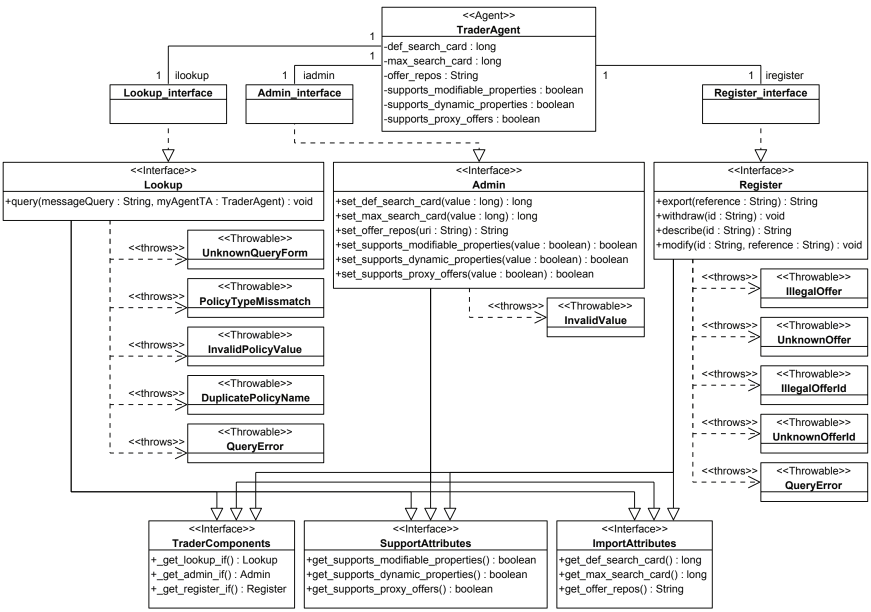


Fig. 1 Trading service class diagram

The *Lookup* interface implements only the action “query”. This operation makes it possible to interrogate the trader about the content of the information in the EID documents. Through the *Admin* interface, trader policy attributes can be modified. The *export*, *modify*, *withdraw* and *describe* operations are accessible by making use of the *Register* interface. For example, *withdraw* is used to delete a document from the trader repository, identifying the document in question.

The object with the "trader" role (TraderAgent), is in charge of registering the EID documents from the ResourceAgent agents (exporters), and returning the EID documents required by QueryAgent (importers). ResourceAgent agents have the role of "exporters", and therefore, their function is to register EID documents in the trader repository. On the other hand, the system components performing the role of importer are called QueryAgent, and it is their function to get, through the trader, EID documents or properties satisfying a certain criterion. The role of "trader administrator" is carried out by the IMIAgent agent. Its function is to define, administer and see that the trading agent policy is complied with. Furthermore, this agent is in charge of controlling the rest of the trading subsystem environment. Finally, ResourceAgent agents also perform the role of "service offer", in addition to the role of "exporter", so they take care of maintaining service offers. Figure 2 shows the relationships among the various agents taking part in the trading service. Relationships are used to describe the communication using ontologies (next described).

Table 1 ODP roles in SOLERES system

ODP specification rol	SOLERES agent name	Description
Trader	Trader Agent	Register and return service offers
Exporter	ResourceAgent	Register service offers in the trader
Importer	QueryAgent	Get service offers from the trader
Trader administrator	IMIAgent	Define and administrate the trasder's policy
Service offer	ResourceAgent	Manage the service offers

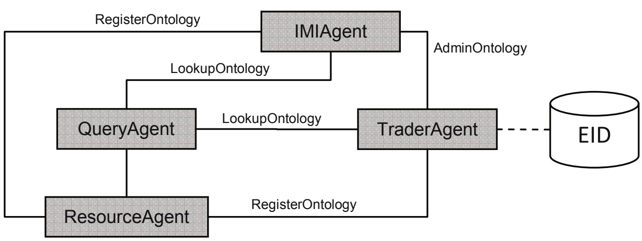


Fig. 2 Agent architecture of SOLERES-KRS

3 Trading Ontologies

Ontologies are used in two very different contexts in SOLERES [9], [10]. The first refers to how offers managed by the trader are represented. On the other hand, ontologies are used to describe queries/answers in communications among the various agents that intervene in the trading service. The trader manages a repository of EID documents. These documents are described in Web Ontology Language (OWL). When the trader requires this information to be manipulated, the corresponding action is expressed in SPARQL language (`http://www.ual.es/acg/soleres/awt`).

The second way in which our system makes use of ontologies refers to the strategy used in implementing trader agent communication with the rest of the system agents. Figure 2 shows the different ontologies used. These ontologies contain the descriptions and semantics of the processes by which the agents can interact with each other. OWL was also used for their construction. Each of these process ontologies corresponds to one of the functional interfaces implemented in the trading service (Lookup, Register, Admin). They were defined based on the methods and attributes included in the implementation of our trader agent. The classes that comprise the process ontologies inherit three main classes: (a) *action*, which represents each of the operations the trader offers by the corresponding interface, (b) *concept*, which corresponds to the input and/or output parameters the action needs to be executed; and (c) *predicate*, which represents the possible results of the action executed. Table 2 shows a partial description of the Register process ontology (export).

Table 2 Register Ontology description

Entity	Assertions
Offer	(uri exactly 1) and (subclass of Concept) and ((Offer_Describe exactly 1) or (Offer_Export exactly 1) or (Offer_Mofidy exactly 1))
Offer_Id	(id exactly 1) and (subclass of Concept) and ((Offer_Describe exactly 1) or (Offer_Export exactly 1) or (Offer_Modify exactly 1) or (Offer_Withdraw exactly 1))
Export	(subclass of Action) and (Export_Offer exactly 1) and (Export_OfferId exactly 1) and ((Export_DuplicateOffer exactly 1) or (Export_ExportedOffer exactly 1) or (Expor_IllegalOffer exactly 1) or (Export_UnknownOffer exactly 1))
Message	(returned_message exactly 1) and (subclass of Predicate)
DuplicateOffer	(subclass of Message) and ((DuplicateOffer_Export exactly 1) or (DuplicateOffer_Modify exactly 1))
ExportedOffer	(subclass of Message) and (ExportedOffer_Export exactly 1)
IllegalOffer	(subclass of Message) and ((IllegalOffer_Export exactly 1) or (IllegalOffer_Modify exactly 1))
UnknownOffer	(subclass of Message) and ((UnknownOffer_Export exactly 1) or (UnknownOffer_Modify exactly 1))

For example, the entity that represents the export operation (Export) is a subclass of the `Action` entity. It also has two associated concepts, `Offer` and `Offer_Id`. With the `Export_Offer` and `Export_OfferId`, respectively. `Offer` contains the information on the route where the document is to be found and `OfferId` on its identification number. The rest of the properties of the `Export` entity correspond to any predicates that may arise from an action when it has been executed, such as, `ExportedOffer` to indicate that the document has been correctly inserted in the repository, `DuplicateOffer` to indicate that the document is already in the repository and `UnknownOffer` to indicate that it could not get the document from the route indicated by the `Offer` concept. To insert a new EID document, the ResourceAgent ("exporter") must make the call to the `export` operation from the trader Register interface. This call to the trader is done by sending a message to the trading agent in which the content must be an instance of the Register process ontology with the elements corresponding to this action.

To implement the SOLERES multiagent system, we used the JADE platform and followed the instructions of the FIPA standard for the communications languages, protocols and interfaces used. FIPA standard communication is done by exchanging Agent Communication Language (ACL) messages between agents. Our system uses OWL as the content language to express communication actions with each of the process ontologies. OWL was chosen because: (a) the knowledge representation expression level is suitable for an agent-based system, (b) treatment and manipulation of the various ontologies is simpler than with the rest of the languages proposed as content for ACL messages, (c) it is a semantic language designed for WIS and web-based services, and (d) it is computer-interpretable, making it possible to execute reasoning and verification tasks on messages exchanged by the agents.

4 An Example of Execution

This example describes the communication process that takes place in the trading service when a system agent makes use of one of the trader operations; a system scenario using a partial class diagram (Figure 3) in which agents and behavior participate starting out from a query for insertion of a new document in the repository.

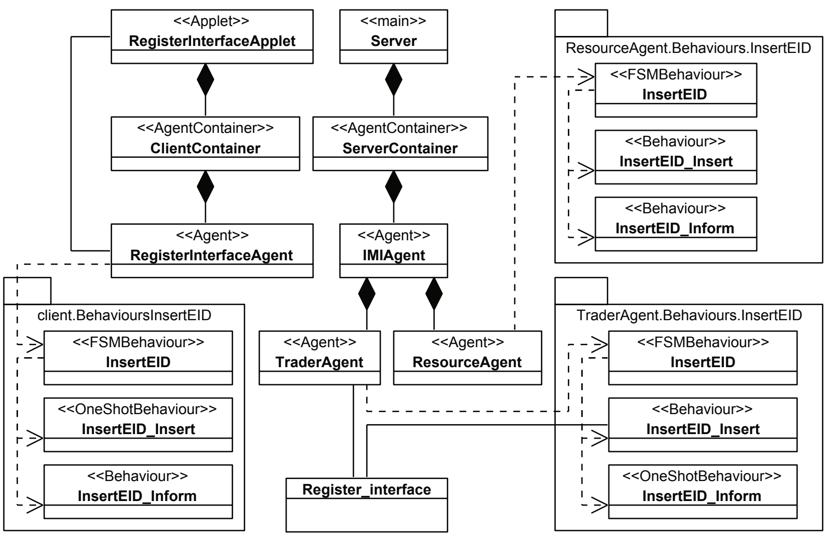


Fig. 3 Example scenario class diagram

In this scenario, a client user, through a GUI agent, executes the trader "export" action. This operation consists of a series of steps: (1) the graphical interface (RegisterInterfaceApplet), allows the user to start the EID document insertion procedure, (2) the GUI agent (RegisterInterfaceAgent) manages the events produced by the user, (3) the interface agent, by its behavior (InsertEID), communicates with the resource agent (ResourceAgent) by means of a message in which it indicates the action it desires, (4) ResourceAgent (exporter) expresses its desire to add an EID document to the trader (TraderAgent) in a message with content expressed in the Register ontology, (5) the TraderAgent executes the action and returns the result to ResourceAgent in a message expressed in the Register ontology, (6) and finally, the ResourceAgent responds to the interface agent with the result to inform the user.

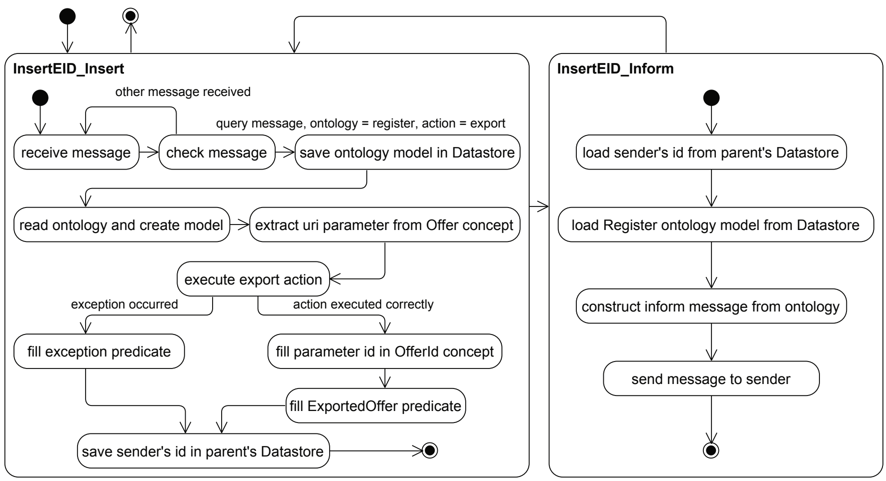


Fig. 4 TraderAgent behavior for insertion of an EID

The subtasks related to execution of the action by the trader are (Figure 4): (a) behavior of the trader in charge of inserting the EID document starts by creating the trader object and waiting to receive a message from another agent which fulfills the Register ontology and the action to be done is *export*; (b) the instance of Register ontology is extracted from the message received; (c) the route for the new document to be inserted is received from the instance of the Offer concept; (d) making use of the Register interface, the *export* action is executed; (e) if the action successful, an instance of the ExportedOffer predicate is created and also the OfferId so the identifier of the document inserted can be returned; othercase, an instance of the predicate corresponding to the exception thrown is created; (f) finally, the new instance of the Register ontology is sent to the Register agent who sent the request for insertion.

Table 3 shows the set of messages exchanged by three system agents that intervene in the EID document insertion action. Finally, the content of the OWL code in the messages pertaining to the Register ontology is shown in Table 4. In this example the message requesting insertion of the EID document, the Export action only has one concept, Offer, where the route for the new document to be inserted is given. In the answering message, the export action has an additional concept, OfferId, where it indicates the identifier of the document inserted, as well as a predicate, ExportedOffer, stating that the action has been correctly carried out.

Table 3 Exchanged messages in the example scenario

Source agent	Target agent	Ontology	Language	Content
RegisterInterfaceAgent	ResourceAgent	—	export	new_EID.owl
ResourceAgent	TraderAgent	Register	export	[message1]
TraderAgent	ResourceAgent	Register	export	[message2]
ResourceAgent	RegisterInterfaceAgent	—	export	EID exported. ID = 1234

Table 4 Content of Register Ontology messages

Message1 content

```
<Export rdf:ID="Export_1">
  <Export_Offer>
   <Offer rdf:ID="Offer_1">
    <uri>new_EID.owl</uri>
   </Offer>
  </Export_Offer>
</Export>
```

Message2 content

```
<Export rdf:ID="Export_1">
  <Export_OfferId>
   <OfferId rdf:ID="OfferId_1"> <id>1234</id>
</OfferId>
  </Export_OfferId>
  <Export_ExportedOffer>
   <ExportedOffer rdf:ID = "ExportedOffer_1">
    <returned_message> EID exported </returned_message>
   </ExportedOffer>
  </Export_ExportedOffer>
  <Export_Offer>
   <Offer rdf:ID="Offer_1"> <uri>new_EID.owl</uri>
   </Offer>
  </Export_Offer>
</Export>
```

5 Conclusions and Future Work

This paper describes implementation of the SOLERES trading service. Following the ODP trading function specification, a conventional trading service has been adapted for use in our information system. In this system, the service offers managed by the trader have been replaced by offers of documents that contain information related to the environmental information system and the trader interface actions have been adapted for their manipulation. These offers have been modeled through ontologies and implemented in the OWL language. To manipulate this information, the SPARQL language is used. Furthermore, the use of ontologies has been extended to model communications between the trader and the rest of the components. In this sense, each of the trading function interfaces has an ontology in the system with the same name which defines the semantics of interaction with the trader. Thus, a component that wishes to communicate with the trader to use this interface must do so in messages that comply with the corresponding ontology.

The SOLERES system is a multiagent system, so each of the components, including the trader, are encapsulated in software agents. This facilitates modeling and implementation of their behavior and also offers additional possibilities for message exchange among system components. In the future, the trading service implemented will be extended to make the prototype able to communicate with other trading agents by adding on Link and Proxy interfaces (trader federation).

Acknowledgments. This work has been supported by the EU (FEDER) and the Spanish MICINN under grant of the R&D project TIN2007-61497. `http://www.ual.es/acg/soleres`.

References

1. ITU/ISO, ODP Trading Function Specification. ISO/IEC International Standard 13235 (1997)
2. Craske, G., Tari, Z., Kumar, K.R.: DOK-Trader: A CORBA Persistent Trader with Query Routing Facilities. In: Proc. of the Int. Symp. on Distributed Objects and Applications, p. 230 (1999)
3. Merz, M., Müller, K., Lamersdorf, W.: Service Trading and Mediation in Distributed Computing Systems. In: Conf. on Distributed Computing Systems, p. 450 (1994)
4. Busse, S., Kutsche, R.D., Leser, U., Weber, H.: Federated Information Systems: Concepts, Terminology and Architectures. Technical Report 99-9, Technical University of Berlin (1999)
5. Lamparter, S., Schnizler, B.: Trading Services in Ontology-driven Markets. In: Proc. of the 2006 ACM Symp. on Applied Computing, pp. 1679–1683 (2006)
6. Tsai, W.T., Huang, Q., Xu, J., Chen, Y., Paul, R.: Ontology-based Dynamic Process Collaboration in Service-Oriented Architecture. In: Proc. Service-Oriented Comp. & App., pp. 39–46 (2007)
7. Collins, J., Ketter, W., Gini, M.: Flexible decision control in an autonomous trading agent. Electronic Commerce Research and Applications (2009)
8. Ziming, Z., Liyi, Z.: An integrated approach for developing e-commerce system. In: Proc. of the Wireless Communications, Networkig and Mobile Computing, pp. 3596–3599 (2007)
9. Asensio, J.A., Iribarne, L., Padilla, N., Muñoz, F.J., Ayala, R., Cruz, M., Menenti, M.: A MDE-based Satellite Image Ontology for Environmental Management Systems. Springer, Heidelberg (2009) (in press)
10. Padilla, N., Iribarne, L., Asensio, J.A., Muñoz, F.J., Ayala, R.: Modelling an Environmental Knowledge-Representation System. In: Lytras, M.D., Carroll, J.M., Damiani, E., Tennyson, R.D. (eds.) WSKS 2008. LNCS (LNAI), vol. 5288, pp. 70–78. Springer, Heidelberg (2008)
11. Iribarne, L., Padilla, N., Asensio, J.A., Muñoz, F.J., Criado, J.: Involving Web-Trading Agents & MAS: An Implementation for searching and recovering environmental information. In: ICAART 2010 (2010) (in press)

An Agent-, Service-Based Platform Supporting Ontological Integration of Proficient Knowledge Managed in Decision Making

Elsa Trigueros, María V. Hurtado, Kawtar Benghazi, and José-Manuel Zurita

Abstract. Case-based reasoning (CBR) is a problem-solving strategy which tries to leverage past experiences (cases) to solve incoming situations. On the other hand, ontologies are proving to be a means to facilitate sharing and reuse of bodies of knowledge across organizations and applications on the basis of a well-defined semantics for concepts and their relationships. To a certain extent, organizations may wish to operate in an autonomous, independent way, but also may wish to improve enterprise collaboration for competitiveness reasons, strategic alliances, etc. In order to achieve the integration of distributed knowledge bases used in decision making systems, a formal characterization of cases through their ontological description is presented, as well as their interrelationships with cases stored in other different repositories. We propose the use of domain classification and term mappings techniques, in concert with an Internet-operative platform that combines agent and service technologies, intended to satisfy and accomplish effective knowledge sharing across organizations.

1 Introduction

In order to make an appropriate decision, decision-makers must not only exhibit competence and motivation, but also have quality information pertinent to the decision at hand [16]. Case-based reasoning is an artificial intelligence technique that refers to past experiences (cases) to help making decisions or taking actions In addition, knowledge sharing requires some kind of technological support for suitable operation.

Elsa Trigueros · María V. Hurtado · Kawtar Benghazi
Departamento de Lenguajes y Sistemas Informáticos, Universidad de Granada, E.T.S.I.I.T., c/ Periodista Daniel Saucedo Aranda s/n, 18071 Granada, Spain
e-mail: {elsa,mhurtado,benghazi}@ugr.es

José-Manuel Zurita
Departamento de Ciencias de la Computación e Inteligencia Artificial, Universidad de Granada, E.T.S.I.I.T., c/ Periodista Daniel Saucedo Aranda s/n, 18071 Granada, Spain
e-mail: zurita@decsai.ugr.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 331–338.
springerlink.com

Nowadays, technologies related to agents, ontologies, cooperative work, etc., have improved knowledge management, sharing and reuse across organizations [4,3]. Agent-based systems have been applied to develop different kinds of CBR systems. For instance, cooperating agents may utilize CBR systems so as to accomplish decision making activities within high level tasks such as system control, simulation and monitoring [2], or a multi-agent system (MAS) can carry out a specific task which could correspond to a concrete stage of a CBR system such as intelligent information retrieval from heterogeneous distributed sources [10]. Web Services is another emerging technology that allows loosely coupling between organizations and provides an important means for reuse of functionality under a composite model.

In order to reach an integration of distributed knowledge sources managed locally and promote sound intra- and inter-enterprise collaboration, a common approach is to apply advanced techniques for knowledge representation and management, and develop an operation infrastructure based on agents with intermediate mechanisms (e.g., services) that fulfills the goals planned. Special efforts are required for the use of CBR in distributed problems that use physically dispersed knowledge [8, 5, 15].

This paper proposes the application of techniques, in particular domain classification and term equivalences, aimed at guarantying sound integration of information sources for decision making systems based on CBR, and the use of a combination of different technologies (ontologies, Internet, agents and services) intended to make possible a real, quality knowledge sharing between enterprises. The rest of this paper is organized as follows. Section 2 depicts our proposal for describing cases semantically. Section 3 presents the architecture, resulting from a combination of agent and service architectures, for semantic-based access to knowledge during a CBR execution cycle with the support of ontologies. Conclusions are summarized in section 4.

2 Semantic Case Description

The reasoning cycle of a CBR comprises four stages: retrieve, reuse, review, and retain. Typically a case comprises a problem and the derived solution to that problem. In this work, we focus on the structural CBR (SCBR) approach, which relies on cases that are described by pre-defined attribute-value pairs [1]. We also use an ontological formalization of case structures in the OWL language that allows the semantics associated with the knowledge domain to be explicitly specified and reasoning capabilities for inferring implicit knowledge to be provided.

In previous work [7], an OWL-based metacase representation which extends the classical <*problem,solution*> structure was presented. Figure 1 shows a diagrammatic representation obtained with the Jambalaya plugin for Protégé [9] of a domain ontology to represent cases. Domain specific cases are described as instances of the previous ontological case metamodel. According to this metamodel, at a general level a case is characterized by a *Problem_statement*, a

Domain, a *Description*, a set of *Reparation_Rules* and a *Solution*. This metamodel is to be instantiated in particular ontologies for specific CBR systems. For instance, Figure 2 shows two sample domain subontologies for two different CBR systems that we have called "*CBR A*" and "*CBR B*", both from the *Finance-Banking* domain.

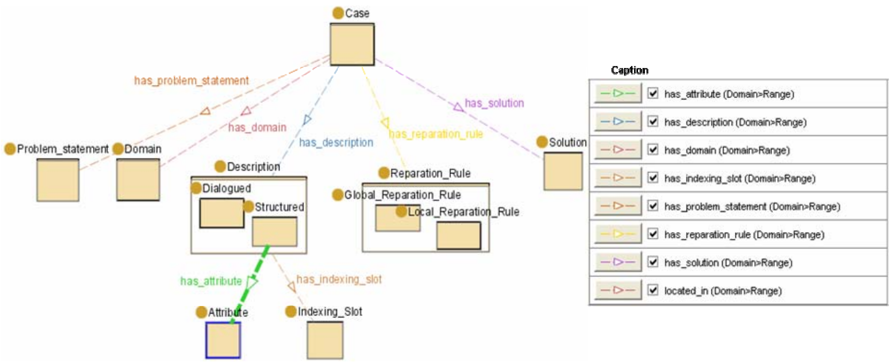


Fig. 1 Ontological metacase

Let us consider that CBR A belongs to a branch office and it stores cases with data about *Financial_Operations* that can be classified into *Liabilities_Operations* and *Assets_Operations*. Examples (instances in the ontology) of *Liabilities* operations are those related to *Investment_Funds*, *Current_Accounts* and *Saving_Accounts*. In turn, examples of *Assets_Operations* are *Financial_Discounts*, *Bank_Guarantees*¸ *Credits* and *Loans*, which can also be subclassified into *Personal_Loans* and *Mortgages*. Likewise, *CBR B* belongs to a company whose business is focused on mortgages and therefore manages a simpler ontology about the domain of cases. Therein, it only distinguishes between *Business_Operations* and *Real_Estate_Operations*.

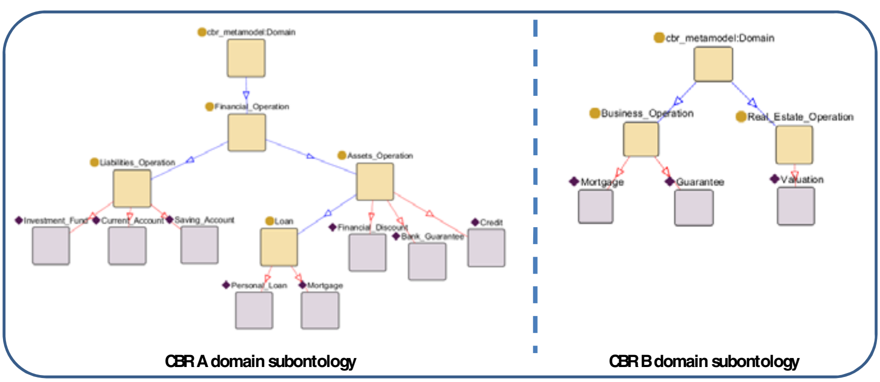


Fig. 2 Two domain subontologies of CBR systems *A* and *B*

3 Technological Platform Supporting Knowledge Integration and Sharing

Knowledge sharing requires agreeing upon basic information representation and management models and techniques, but also an advanced technological support for an effective interoperation. In this section, we consequently propose a platform based in a combination of agent and service architectures, as well as specific support mechanisms that make possible the collaboration between autonomous CBR systems (from different organizations) in order to improve decision making. In the context of the present work, various organizations and actors use knowledge bases implemented by means of several CBRs for a particular domain. They use a similar vocabulary and want to cooperate. The following subsections describe the techniques and platform supporting the previously planned knowledge integration.

3.1 Identification of Actors and Functional Subsystems

There exist three different types of actors: the *administrator*, the *user* and the *expert*; and two different types of agents: the *user agent* and the *CBR agent*.

The system is split into three different subsystems:

- *CBR node* allows the CBR agent to perform the following operations: resolve a case, contact with other CBR agents, store the solution case, learn, retrieve a case, check an unknown term, make the case adaptation and select the best case to provide a solution.
- *User node* allows the user to introduce the problem to be resolved. Additionally, this node lets the expert evaluate a given solution and repair it by modifying the solution or changing the value of a similitude function. In this node, the user agent can invoke the CBR node to start the decision making process.
- *Term Generation node* allows the administrator to add or remove knowledge bases and also the expert to introduce new terms in the system.

3.2 Architecture

The platform is based on an agent architecture consisting of a set of two kinds of agents: user agents and CBR agents (see Figure **2**). The user agent is in charge of maintaining the interaction about requests from a specific expert or user. User agents abstract users from the detailed configuration and operation of the system. User agents serve as a communication means between users and knowledge bases providing transparency at different levels: location of CBR agents, fault tolerance in communication and processing, and task balancing for performance. Thus, user agents can better decide CBR agent candidates to be contacted when user's requests need to be processed, and thereby fostering transparency, since the final user ignores how the making-decision process is carried out.

CBR agents play different roles related to the four phases of a CBR execution process. Thus, each one implements an autonomous CBR system locally, including

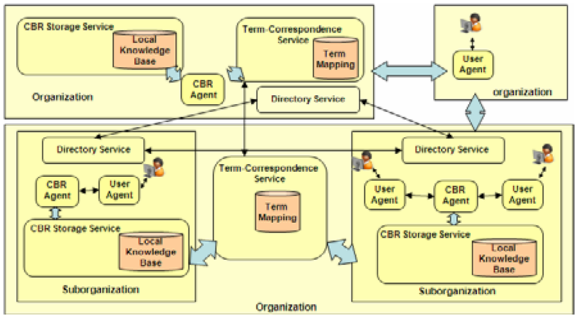


Fig. 3 System architecture

local knowledge space for an organization. Moreover, CBR agents can cooperate with each other: seeking and recovering related cases in other knowledge bases and storing term equivalences for attributes or domains in other remote repositories.

Loosely coupling is obtained thanks to services. There is a direct connection between the roles that agents play and the services they use [13]. Depending on the service an agent invokes, the agent will play a different role. In particular, the user agent plays either a *consultant* role with respect to the directory service, or an *initiator* role wrt the reasoning service (which is implicit in the CBR agent), or a *modifier* role wrt the Term-Correspondence service. The CBR agent plays a *consultant* role wrt the Term-Correspondence service. Thereby, the system does not depend on either the implementation platform, or the data which are being used, or who is invoking the system, which provides a high grade of interoperability and extensibility.

3.3 Service-Based Agent Coordination

As for the location of companies, a user agent uses the directory service, through which it obtains the connection parameters of the CBR agents which act in a certain domain. Once the user agent has obtained the list of close CBR agents, it tries to connect with all of them in order to start the decision making process. Each CBR agent supports a record about the domains it is acting in.

Furthermore, there exist equivalences, called "Term Mappings", that maintain correspondences between the terms used by different CBR systems. Thus, term mapping are to be used in two ways during the case retrieval stage. First, when an agent is searching for cases of a particular domain. For example, as shown in figure 2, two CBR systems may hold different classifications about the same or closely related domains. In this situation, term mappings may help to identify cases of the same domain. Then, term mappings may also be used to compute the degree similarity between two case descriptions from the correspondences between the terms used in such descriptions.

At the right-hand side of figure 4 it is shown an example of a term mapping ontology between the domain ontologies of CBR A and CBR B. The mapping establishes the correspondences between *Assets_Operation* and *Business_Operation*

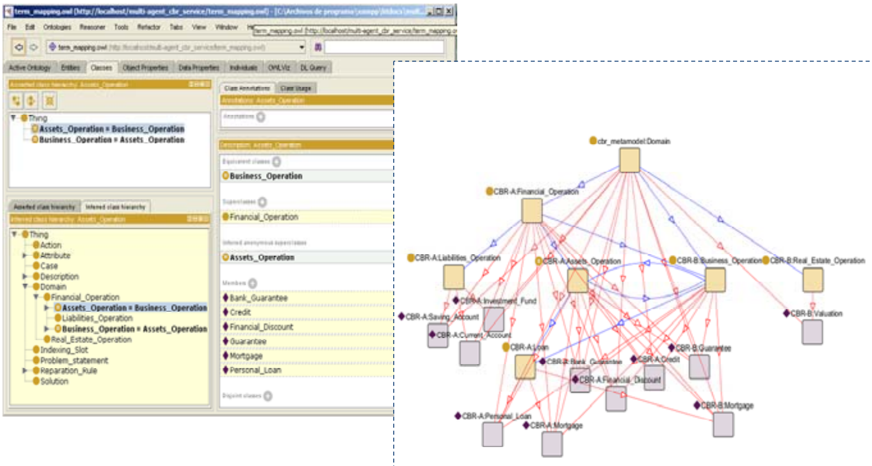


Fig. 4 Example of term mapping between CBR A and B domain ontologies, and graphical representation of the resulting domain ontology

classes. As a result of this mapping, a reasoner would be able to infer that, for example, a *Credit* (instance defined in the CBR A domain ontology only) is a *Business_Operation* (type defined in the CBR B domain ontology only).

At the left-hand side of figure 4 it is shown a graphical representation of the resulting ontology. When a new case C_C arrives the system, the term mapping repository allows CBR agents to look for into other CBR case bases with the aim of obtaining a good case with which resolve the current problem. The relationship between these two terms should be stored in the term mapping repository.

Term mappings are created by experts and then maintained automatically by the Term-Correspondence service. Each CBR agent has associated its term mapping information for reasons of efficiency and fault tolerance (the information is replicated). At the beginning of the decision making process, involved CBR agents send this information to its Term-Correspondence service which looks for new terms or relations and supplements the new term mapping information.

3.4 *Quality Architecture Attributes*

Agent technology has already been successfully applied to CBR in a wide range of application domains and in different ways [7]. CBR systems could be categorized depending on [11]: (1) they make use of a single or multiple agents, and (2) they manage knowledge from a single or multiple case bases. The presented proposal is placed on the Multiple Agents/Multiple Case Bases setting since they help to fulfill the general requirement of supporting autonomous and cooperative work. Additionaly, specific design decisions, which are embodied in the architecture, have been taken so as to achieve other quality attributes:

- Scalability: the software does not need to change when the system grows in size. That is because new CBR agents also cooperate by giving solutions in particular domains according to the information added to the directory service. Moreover, directory and term mapping services are replicated thereby avoiding frequent bottlenecks in a centralized configuration.
- Fault-tolerance: the system neither produces incorrect results nor stops when an error occurs (i.e., a CBR agent fails) because CBR agents themselves collect the solutions provided at a given moment, and agree on a unique solution. Thus, the system operation does not depend on a particular CBR agent. In case of error, only the solution to be provided by the failed CBR agent is affected Additionally, replicated services used in our approach, avoid partial failures (failures of some components), which are common in distributed systems.
- Transparency: The occultation of several aspects such as component location, concurrency, replication, failures and migration, from the user viewpoint. For instance, the system user interacts with the user agent using the directory service which is in charge of looking for the CBR agents that manage case bases for the same domain.

In this way, the architecture follows the paradigm of a cooperating process group that consists of the distribution and replication of components (agents, services and knowledge). This setting exploits concurrency and improves tolerance to faults.

4 Conclusions

This paper starts from the ontological formalization of case structure by means of the OWL language, which contributes to a precise integration of distributed case bases thanks to agreements about a general structure for all cases and a common terminology for domains organized hierarchically. Thereby, reasoning phases of CBR systems benefit in terms of applying advanced similarity measures for retrieval, new kinds of adaptation rules and powerful learning mechanisms. Moreover, OWL enhances the expressiveness, support inference, knowledge reuse, and reasoners allow certain verification procedures, such as consistency check, concept satisfiability, classification and realization to be carried out [12]. Further capabilities on this issue will be studied and exploited in future work.

A platform based on a combination of agent and service architectures is proposed to give coordinated support to expert integration and sharing of experiences acquired in several organizations. It incorporates the mechanisms, i.e., communication and cooperation between agents for problem solving and dynamic management of the knowledge through term correspondences, that make possible such integration and sharing. In addition to achieve a very decoupled and interoperable system, the design decisions adopted aim to satisfy other important quality properties such as scalability and fault tolerance.

So far, a fully Internet operable platform has been developed so as to evaluate the feasibility of the methods proposed herein. Preliminary tests exhibit slow refresh latencies and somehow high response times in the calculation of transitive equivalences between terms. At present, we are working in the design of exhaustive tests in order to validate the applicability of our proposal in real world situations.

Acknowledgments. This research is supported by R+D projects of the Spanish MCYT under project TIN2007-60199.

Referentes

1. Bergmann, R., Schaaf, M.: On the Relations between Structural Case-Based Reasoning and Ontology-based Knowledge Management. Journal of Universal Computer Science 9(7), 608–626 (2003)
2. Budinska, I., Frankovic, B., Dang, T.T.: Intelligent Agents in Support System for Discrete Process Control Design. In: Proc. International Conference in Memoriam of John von Neumann, pp. 23–34 (2003)
3. Castro, J.L., Sánchez, J.M., Zurita, J.M.: Simplifying Case Retrieval in Case-based Reasoning using Ontologies. Artificial Intelligence and Soft Computing (2005)
4. Castro, J.L., Navarro, M., Sanchez, J.M., Zurita, J.M.: Default reasoning with ontologies. In: Procc. Intelligent Systems and Agents, pp. 35–43 (2007)
5. Chaudhury, S., Singh, T., Goswami, P.S.: Distributed fuzzy case based reasoning. Applied Soft Computing 4, 323–343 (2004)
6. Davies, J.: Towards the Semantic Web, Ontology-Driven Knowledge Management. John Wiley & Sons, Chichester (2002)
7. Garrido, J.L., Hurtado, M.V., Noguera, M., Zurita, J.M.: Using a CBR Approach based on Ontologies for Recommendation and Reuse of Knowledge Sharing in Decision Making. In: 8th International Conference on Hybrid Intelligent Systems (HIS 2008), pp. 837–842. IEEE Press, Los Alamitos (2008)
8. Hayes, C., Cunningham, P., Doyle, M.: Distributed CBR using XML (Department of Computer Science Trinity College, Dublin) (2001)
9. Knublauch, H., Fergerson, R.W., Noy, N.F., Musen, M.: The Protégé OWL Plugin: An Open Development Environment for Semantic Web Applications. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) ISWC 2004. LNCS, vol. 3298, pp. 229–243. Springer, Heidelberg (2004)
10. Pelletier, S., Pierre, S., Hoang, H.: Modeling a Multi-Agent System for Retrieving Information from Distributed Sources. Journal of Computing and Information Technology CIT 11, 15–39 (2003)
11. Plaza, E., McGinty, L.: Distributed Case-based Reasoning. Knowledge Engineering Review 20, 261–265 (2006)
12. Sirin, E., Parsia, B., Cuenca-Grau, B., Kalyanpur, A., Katz, Y.: Pellet: A practical OWL-DL reasoner. Journal of Web Semantics 5, 51–53 (2007)
13. Specification for the UML Profile and Metamodel for Services (UPMS) (April 16, 2009), `http://www.omg.org/spec/soaml/1.0/`
14. W3C Web Ontology Working Group, `http://www.w3.org/2001/sw/WebOnt/`
15. Watson, I., Gardingen, D.: A Distributed Case Based Reasoning Application for Engineering Sales Support. In: Procc. of the IJCAI 1999, pp. 600–605 (1999)
16. Yates, J.F., Tschirhart, M.D.: Decision Making Expertise. In: Ericsson, A., et al. (eds.) Cambridge Handbook on Expertise and Expert Performance. Cambridge Univ. Press, Cambridge (2006)

A Multi-agent Recommender System to Suggest Documents in Communities of Practice

Aurora Vizcaíno, Juan Pablo Soto, Javier Portillo-Rodríguez, and Mario Piattini

Abstract. The importance of knowledge management has, in recent years, led to the incorporation of Knowledge Management Systems (KMS) into companies. Some of these KMS could be considered as Recommender Systems that are able to recommend knowledge, which is part of the company's intellectual capital. However, these KMS are not always welcome in the company, since the knowledge is not stored by using a quality control, or because employees feel that these kinds of systems, rather then helping them, cause them extra work. In this paper we present an agent architecture combined with a trust model trying to avoid some of the problems that appear when a KMS is introduced into companies.

Keywords: Knowledge Management Systems, Recommender Systems, Agent Architecture.

1 Introduction

In recent years, knowledge has become an extremely important factor, to the point that intellectual capital is now one of the most important assets for many organizations [1]. At present organizations must operate in a climate of rapid market change and high information volume which increases the necessity to create Knowledge Management Systems (KMS) that support the knowledge process. It is possible to consider certain Recommender Systems, such as KMS, which recommend knowledge, information or documents to employees with the goal of reusing a company's intellectual capital [2].

Aurora Vizcaíno · Javier Portillo-Rodríguez · Mario Piattini
Alarcos Research Group,
Universidad de Castilla – La Mancha
Paseo de la Universidad, 4, 13071, Ciudad Real, Spain
e-mail: {aurora.vizcaino,javier.portillo,mario.piattini}@uclm.es

Juan Pablo Soto
Departamento de Matemáticas,
Universidad de Sonora
Rosales s/n, Centro, 83000, Hermosillo, México
e-mail: jpsoto@gauss.mat.uson.mx

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 339–346.
springerlink.com

However, these kinds of systems are not always welcomed by a company's employees because [3]:

1. Employees often feel that they are overloaded with new work as they have to introduce Knowledge Objects (KOs) into the system.
2. On other occasions employees waste a considerable amount of time searching for information since they do not know the best way in which to do so or they do not know what information may be useful.
3. With regard to the previous point, sometimes there is not quality control with regard to the KOs introduced into the system, and when employees finally discover information or review the information recommended they realise that it is not useful.
4. Employees may introduce information into the systems which is not very valuable, with the sole goal of appearing to be contributing since some companies give incentives to employees that contribute with knowledge.

Our work is focused on attempting to reduce the impact of these last three problems. We therefore use software agents to search for information on behalf of users, and these software agents are in charge of recommending the most suitable knowledge to them. In order to tackle problems three and four we have developed an internal agent architecture and a trust algorithm with which to rate KOs and Knowledge Sources (KSs). The software agents will therefore use this algorithm in order to decide whether a KO or KS should be recommended to a particular user.

The agent architecture and the trust model are focused on Communities of Practice (CoPs) which are a natural means of sharing knowledge. CoPs enable their members to benefit from each other's knowledge. This knowledge resides not only in people's minds but also in the interaction between people and documents. CoPs share values, beliefs, and ways of doing things. Therefore, many companies report that such communities help to reduce the problems caused by a lack of communication, and save time by 'working smarter' [4].

Therefore in Section 2 the architecture is described and later, in Section 3 the recommender system developed in order to check the efficiency of our proposals is explained. After that in Section 4, a brief description of the algorithm used in the architecture to check whether a KO or KS is trustworthy is outlined. Finally, our conclusions are outlined in Section 5.

2 Internal Agent Architecture

The internal agent architecture proposed is composed of two levels: reactive and social-deliberative.

The reactive level is considered by other authors to be a typical level that a Multi-agent System (MAS) must have [5]. A deliberative level is often also considered as a typical level, but a social level is not often considered in an explicit manner, despite the fact that these systems (MAS) are composed of several individuals, the interactions between them and the plans constructed by them. The social level is only considered in those systems that attempt to simulate social behavior. Since we wish to emulate human feelings such as trust and intuition, we have added a social level that considers the social aspects of a community and which takes into account the opinions and behavior of each of the members of that community. Other previous works have also added a social level. For example, in [6] the authors attempts to emulate human emotions such as fear, thirst or bravery, but the architecture used is made up of three levels: reactive, deliberative and social. In our case the deliberative and social levels are not separate levels since we are aware that plans created in the deliberative level involve social interactions. We therefore consider that, in our case, it might be more efficient to define a level which is composed of two parts (social-deliberative level) rather than considering two separate levels. Each of these levels is explained in the following subsections.

2.1 Social-Deliberative Level

At this level, the agent has a type of behavior which is oriented towards objectives, that is, it takes the initiative in order to plan its performance with the purpose of attaining its goals. The components of the social-deliberative level are (see Figure 1):

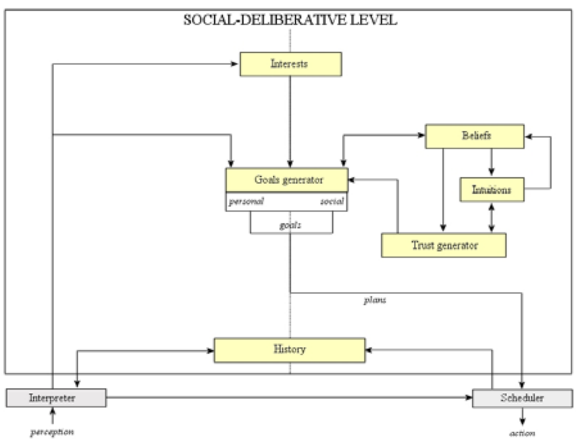


Fig. 1 Deliberative-Social level

Goals generator. Depending on the state of the agent, this module must decide what the most important goal to be achieved is.

Social beliefs. This component represents a view that the agent has of the communities and their members, for instance, beliefs about other agents.

Social interests. This is a special type of belief. In this case it represents interest in other agents.

Intuitions. As we are modelling community members we have attempted to introduce factors into this architecture that influence people when they need to make decisions about whether or not to trust in a knowledge source. One of these factors is intuition, which is a subjective factor since it depends on each individual. This concept is extremely important when people do not have any previous experience. Other authors have called this issue "indirect reputation or prior-derived reputation" [7]. In human societies, each of us probably has different prior beliefs about the trustworthiness of strangers we meet. Sexual or racial discrimination might be a consequence of such prior belief [7]. We often trust more in people who have similar features to our own. For example, when a person consults a community in a search for rating products or services such as *Tripadvisor* [8], s/he often checks comments from people who are of the same age as or have similar interests to him/her. In this research, intuition has therefore been modelled according to the similarity between agents' profiles: the greater the similarity between one agent and another, the greater the level of trust. The agents' profiles may change according to the community in which they are working.

Trust generator. This module is in charge of generating a trust value for the knowledge sources with which an agent interacts in the community. To do this, the trust generator module considers the trust model explained in detail in [9] which considers the information obtained from the internal model and the agent's intuitions.

2.2 Reactive Level

This is the level in charge of perceiving changes in its environment and responding to these changes at the precise moment at which they occur, i.e., when an agent executes another agent's request without any type of reasoning. The components that form part of the reactive level have been omitted due to space constraints.

3 A Recommender System

A recommender system has been developed in order to test the internal agent architecture. In this system each CoP member is represented by a software agent called a User Agent. A new community member must first join a community, which is done by using the "Register" menu and choosing a community from those which are available. Once registered, a member can provide new KOs or use those which are already available in the community and/or propose new subjects. The two first situations are described.

1. **Proposing a new KO.** In order to provide a KO (for instance a document) a person must use the "Propose" menu and must configure the following options:

- *Topic*: There may be different topics or areas in each community. The users can choose that in which they intend to propose the document.
- *Select Document*: The proposed KO (in this case a document).

Once the user has chosen the options, the User Agent sends the values to another software agent called the Manager Agent which is in charge of adding the new KO to the community and modifying the frequency of contribution of the User Agent in that community.

2. **Using community KO.** Members can request the recommendation of a particular KO and their User Agent will help them to find that which is most suitable. Therefore, when a person searches for a KO relating to a particular subject their User Agent consults the Manager Agent about which KOs are related to that subject. The Manager Agent then replies with a list of KOs. The User Agent sorts this list (list on left of Figure 2) by using an algorithm which will be explained in the following section of this paper. The User Agent can therefore detect how worthy a KO is, thus saving employees time, since they do not need to review all the KOs related to a particular subject but only those considered to be most relevant by the members of the community or by the user him/herself.

Once one or several KOs have been chosen, the user must then evaluate the KOs consulted in order to provide feedback to the community about them (list on right of Figure 2).

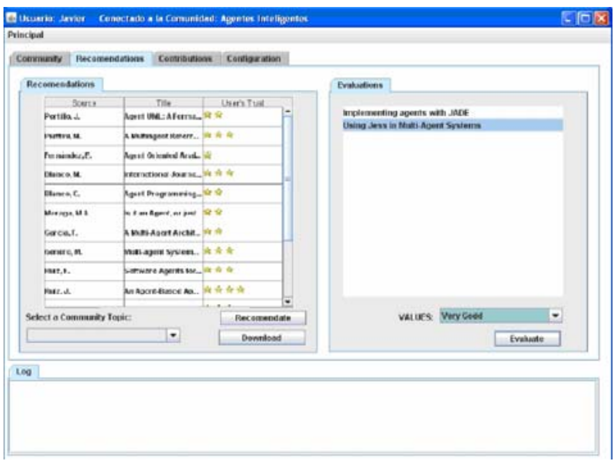


Fig. 2 Using community KOs

4 Description of the Algorithm

This section describes the algorithm used to permit the agents to recommend a KO. The input of this algorithm is a set of KOs. Each KO may or may not have been evaluated previously, signifying that a KO may already have a list of evaluations (along with the identity of each person who evaluated it), or it may not have any evaluations. This aspect will be taken into account by the algorithm, which therefore distinguishes two groups:

Group 1 (G1): This group is formed of the KOs that have already been evaluated. This is the most important group since if the agents have previous evaluations of a KO they have more information about it, which facilitates the task of discovering whether or not its recommendation is advisable.

Group 2 (G2): these KOs have not been used previously so the agents do not have any previous evaluations of them. Let us now observe how each group is processed by the algorithm.

In G1 the KOs will be ordered by a Recommendation Rate which is calculated by the User Agent for each KO. Hence RR_k signifies the Recommendation Rate for a particular KO called k, and is obtained from:

$$RR_k = w_1 \times TE_i + w_2 \times TS_{ik} \tag{1}$$

where TE_i is the pondered mean of the evaluations determined by the trust that an agent "*i*" has in each evaluator (the person who has previously evaluated that KO). TE_i is calculated as:

$$TE_i = \frac{\sum_{j=1}^{n} (E_{jk} \times TS_{ij})}{\sum_{j=}^{n} TS_{ij}} \tag{2}$$

Therefore, TS_{ij} is the trust value that the User Agent "*i*" has in the knowledge source "*j*", since in a CoP the source which provides a KO will usually be a CoP member. TS_{ij} therefore represents the trust that an agent "*i*" has in another agent "*j*" (explained in detail in [11]) and E_{jk} is the evaluation that an agent "*j*" has made with regard to a particular KO "*k*".

The parameter TS_{ik} used in Formula (1) similarly indicates the trust that an agent "*i*" has in a knowledge source "*k*". In other words, the agent must take two things into consideration when calculating the RR_K:

- The other agents' opinions of a KO "*k*" pondered by the trust that agent "*i*" has in the person who provided that evaluation.
- The opinion that the agent "*i*" has in the agent that has provided the KO "*k*".

Both w1 and w2 are weights which are used to adjust the formula. The sum of w1 and w2 should be 1.

G2 will use another formula to calculate the RR_k for each KO since, in this case, there are no results of previous evaluations of the KOs. The formula used is, therefore:

$$RR_k = w_1 \times TS_{ix} + w_2 \times \mathrm{Re}_x \tag{3}$$

where TS_{ix} is the Trust that the User Agent "i" has in the KS "x" which provides the KO "k", and Re_x is the reputation that the KS has (according to another member's agent's opinion). This Re_x value is calculated by asking those agents with a higher trust value about the KS with a weighted mean, which is subsequently calculated. Re_x is therefore obtained as:

$$\mathrm{Re}_x = \frac{\sum_{j=1}^{n} (TS_{jx} \times TS_{ij})}{\sum_{j=1}^{n} TS_{ij}} \tag{4}$$

where TS_{jx} is the trust that an agent "*j*" has in the KS "*x*" and TS_{ij} is the trust value that the agent "*i*" has in agent "*j*". Therefore, the agent's opinion of KS "*x*" is adjusted by the opinion that the agent "*i*" has with regard to the agent that is giving its "opinion" (trust value in the KS "*x*"). It is possible to stress the values of TS_{ix} and Re_x by using the weights w_1 and w_2.

5 Conclusions

CoPs are a means of knowledge sharing. However, the knowledge that is reused should be valuable for its members, who might otherwise prefer to ignore the documents that a community has at its disposal. In order to encourage the reuse of documents in CoPs, in this work we propose a multi-agent recommender system with which to suggest trustworthy documents. Some of the advantages of our system are:

- The use of agents to represent members of the community helps members to avoid the problem of information overload since the system gives the User Agents the ability to reason about the trustworthiness of the other agents or about the recommendation of the most suitable documents to the members of the community. Users are not, therefore, flooded with all the documents that exist with regard to a particular subject, but their User Agents filter them and recommend only those which are most trustworthy (when they have rates) or those which are provided by more trustworthy sources or sources which have preferences and features that are similar to those of the user in question.
- Detecting whether members store documents that are not useful, since the system provides users with the opportunity to evaluate the documents consulted, and when a document is frequently evaluated with low marks then the Manager Agent will check who the provider is and whether most of that person's documents have a low evaluation, two options can be considered. First, that the person does not have sufficient knowledge of the subject, in which case the Manager Agent can consult that person's Level of Expertise (which is indicated when a person joins a community), and modify it if necessary. The second option is that this person may be consciously introducing invaluable documents. In this case the trust in this source will be low and the documents will rarely be recommended. The system can also detect those users with the greatest level of participation and those whose documents have obtained higher rates. This information can be used for two purposes: expert detection and/or recognition of fraudulent members who contribute with worthless documents. Both functionalities imply various advantages for any kind of organization, i.e., the former permits the identification of employee expertise and measures the quality of their contributions, and the latter permits the detection of fraud when users contribute with non-valuable information.
- The system facilitates the exchange and reuse of information, since the most suitable documents are recommended. Furthermore, the proposed algorithm can calculate a trust value even when the community has only recently been created since, in order to calculate trust, various known factors are used such as

Position, Level of Expertise and even Intuition [9]. This is a key difference with regard to other algorithms which use only previous experience and which cannot then calculate trust values if the system is just starting to work. When a new member arrives it is also impossible for other algorithms to calculate a previous trust value related to this new member. A further contribution of this algorithm is that it is quite flexible since in many situations weights are used to modify the formulas. This algorithm could, therefore, be used by the designers of other recommender systems who could decide what values they should give to these weights in order to adapt the formula to their needs.

Acknowledgments. This work is partially supported by the ENGLOBAS (PII2I09-0147-8235) and MELISA (PAC08-0142-3315) projects, Junta de Comunidades de Castilla-La Mancha, Consejería de Educación y Ciencia, in Spain. It is also supported by PEGASO (TIN2009-13718-C02-01), Ministerio de Educación y Ciencia (Dirección General de Investigación)/ Fondos Europeos de Desarrollo Regional (FEDER) and the FABRUM project (grant PPT-430000-2008-063), Ministerio de Ciencia e Innovación in Spain.

References

1. Hansen, B.H., Kautz, K.: Knowledge mapping: A technique for identifying knowledge flows in software organisations. In: Dingsøyr, T. (ed.) EuroSPI 2004. LNCS, vol. 3281, pp. 126–137. Springer, Heidelberg (2004)
2. Barbosa, E., Oliveira, J., Maia, L., de Souza, J.M.: Using recommendation systems for explicit knowledge dissemination and profiling identification for scientific and engineering contexts. In: 11th International Conference on Computer Supported Cooperative Work in Design (CSCWD), vol. 1, 2, pp. 715–721 (2007)
3. Lawton, G.: Knowledge Management: Ready for Prime Time? Computer 34(2), 12–14 (2001)
4. Wenger, E.C., Snyder, W.M.: Communities of practice: The organizational frontier. Harvard Business Review 78(1) (2000)
5. Ushida, Y., Hirayama, Y., Nakajima, H.: Emotion model for life-like agent and its evaluation. In: Fifteenth National Conference on Artificial Intelligence (AAAI 1998) and Tenth Conference on Innovative Applications of Artifical Intelligence (IAAI 1998), pp. 62–69 (1998)
6. Imbert, R., de Antonio, A.: When emotion does not mean loss of control. In: Panayiotopoulos, T., Gratch, J., Aylett, R.S., Ballin, D., Olivier, P., Rist, T. (eds.) IVA 2005. LNCS (LNAI), vol. 3661, pp. 152–165. Springer, Heidelberg (2005)
7. Mui, L., Mohtashemi, M., Halberstadt, A.: Notions of reputation in multi-agents systems: a review. In: Proceedings of the first international joint conference on Autonomous Agents and Multiagent Systems, pp. 280–287 (2002)
8. Tripadvisor, `http://www.tripadvisor.com`
9. Soto, J.P., Vizcaíno, A., Portillo-Rodríguez, J., Piattini, M.: Why Should I Trust in a Virtual Community Member? In: Fonseca, B. (ed.) CRIWG 2009. LNCS, vol. 5784, pp. 126–133. Springer, Heidelberg (2009)

AIDeM: Agent-Based Intrusion Detection Mechanism

Cristian Pinzón, Martí Navarro, and Javier Bajo

Abstract. The availability of services can be comprimised if a service request sent to the web services server hides some form of attack within its contents. This article presents AIDeM (An Agent-Based Intrusion Detection Mechanism), an adaptive solution for dealing with DoS attacks in Web service environments. The solution proposes a two phased mechanism in which each phase incorporates a special type of CBR-BDI agent that functions as a classifier. In the first phase, a case-based reasoning (CBR) engine utilizes a Naïves Bayes strategy to carry out an initial filter, and in the second phase, a CBR engine incorporates a neural network to complete the classification mechanism. AIDeM has been applied within the FUSION@ architecture to improve its current security mechanism. A prototype of the architecture was developed and applied to a case study. The results obtained are presented in this study.

Keywords: Availability, Web Service Attack, Multi-agent, case-based reasoning.

1 Introduction

Security is one of the primary concerns in service oriented architectures (SOA) and Web services [1]. Some protective measures such as Web Service Security (WSS) [2], WS-Policy [3], WS-Trust [4], etc. focus on authorization and authentication aspects to ensure confidentiality and integrity. However, they do not contemplate security problems that put the availability of Web services at risk. The

Cristian Pinzón
Universidad Tecnológica de Panamá, Av. Manuel Espinosa Batista, Panama
e-mail: cristian.pinzon@utp.ac.pa

Martí Navarro
Departamento de Sistemas Informáticos y Computación, Universidad Polítecnica de Valencia, Camino de Veras s/n, 46022, Valencia, España
e-mail: mnavarro@dsic.upv.es

Javier Bajo
Escuela Universitaria de Informática, Universidad Pontificia de Salamanca, Compañia , 5 37002, Salamanca, Spain
e-mail: jbajope@upsa.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 347–354.
springerlink.com

emergence of new threats that can interrupt the correct functioning of services is closely related to some of the components contained in this technology, such as the XML standard used to encode messages, and the hypertext transfer protocol (HTTP) used to the communication. Different types of threats, similar to denial of service (DOS) attacks, can incapacitate a web service and block access to authorized users by sending malicious requests to the web server.

This study presents AIDeM, an advanced detection method that can confront mechanisms or techniques that produce denial of service attacks within Web environments. AIDeM is intended to improve the initial security level within the FUSION@ architecture [5]. FUSION@ proposes a new and easier method to develop distributed intelligent ubiquitous systems, where applications and services can communicate in a distributed way with intelligent agents, even from mobile devices, regardless of time and location restrictions. FUSION@ did already include a security component within its structure consisting of an agent specialized. However, the security method employed by this agent is limited in scope making available services vulnerable to attack. AIDeM is based on a group of agents specially designed to work together intelligently and adaptively to solve the problem of the reliability of SOAP messages sent in service requests. The core of AIDeM is a classification mechanism that incorporates a two-phase strategy to classify SOAP messages. The first phase applies an initial filter for detecting simple attacks without requiring an excessive amount of resources. The second phase involves a more complex process that ends up using a significantly higher amount of resources. Each of the phases incorporates an intelligent agent that integrates a CBR engine with advanced classification capabilities. The idea of a CBR mechanism is to exploit the experience gained from similar problems in the past and then adapt successful solutions to the current problem [6]. The first agent uses a Naïves Bayes classifier and the second a neural network, each of which is incorporated into the respective re-use phase of the CBR cycle. As a result, the system can learn and adapt to the attacks and the changes in the techniques used in the attacks. Additionally, a strategy of a two phased classification mechanism is to use its resources (CPU, cycle, memory) and improve response time.

The rest of the paper is structured as follows: section 2 presents a general description of the FUSION@ architecture and the limitations of the current mechanism of security. Section 3 focuses on the details of the AIDeM architecture and the mechanism of classification. Finally, section 4 describes how the classifier agent has been tested inside a multi-agent system and presents the results obtained.

2 FUSION@ Architecture and Current Mechanism of Security

FUSION@ [5] combines a services-oriented approach with intelligent agents to obtain an innovative architecture that facilitates ubiquitous computation and communication, and high levels of human-system-environment interaction. It also provides an advanced flexibility and customization to easily add, modify or remove applications or services on demand, regardless of the programming language. FUSION@ framework defines four basic blocks: a) Applications represent all the programs that can be used to exploit the system functionalities. They can be executed locally or remotely. b) Services represent the activities that the architecture offers. c) Agents Platform is the core of the architecture and integrates a set of agents, each one with special characteristics and behaviours. In FUSION@ services are managed and coordinated by deliberative BDI agents with distributed computation and coordination abilities. d) Communication Protocol allows applications and services to communicate directly with the agents platform. The protocol is completely open and independent of any programming language, facilitating ubiquitous communication capabilities. This protocol is based on SOAP specification [7] to capture all messages between the platform and the services and applications. Developers are free to use any programming language. The only requirement is that they must follow the communication protocol based on the transactions of XML (SOAP) messages. The communication among agents in the platform follows the FIPA Agent Communication Language (ACL) specification.

FUSION@ is a modular multi-agent architecture, where services and applications are managed and controlled by deliberative BDI (Belief, Desire, Intention) agents [8] [9]. There are different kinds of agents in the architecture, each one with specific roles, capabilities and characteristics:

- CommApp Agent. Responsible for all communication between applications and the platform.
- CommServ Agent. Responsible for all communications between services and the platform. This agent also periodically checks the status of all services to know if they are idle, busy, or crashed.
- Directory Agent. Manages the list of services that can be used by the system.
- Supervisor Agent. Supervises the correct functioning of the other agents in the system.
- Security Agent. Analyzes the structure and syntax of all incoming and outgoing XML messages.
- Admin Agent. Decides which agent must be called by taking the QoS and user preferences into account. Admin Agent has a routing list to manage messages from all applications and services. This agent also checks if services are working properly to ensure that QoS is always current.
- Interface Agent. This particular agent was designed to be embedded in user applications.

Developers can add new agent types or extend the existing ones to conform to their projects needs. However, most of the agents' functionalities should be modelled as services, releasing them from tasks that could be performed by services. More specific details of the architecture can be found in [7].

Security is considered an important element within the FUSION@ architecture. As a result FUSION@ incorporates a security mechanism that validates all incoming messages dealing with service requests. The strategy is centered on the role of the Security agent and its attempt to protect services that are facing potential attacks hidden within service requests (embedded within a SOAP message). The security mechanism contained within the Security agent is simple and efficient for known attacks, although it presents a series of limitations when it comes to protecting the architecture and services during a more complex attack. The majority of the attacks made against service based environments use complex techniques that are difficult to detect with a simple XML code review found in the SOAP message. One example of a complex attack directed at service based environments is the denial of service attack, which can implement many mechanisms of attacks within service based environments [1]. A DoS attack causes the resources available in the server of the provider (memory and CPU cycles) to be drastically reduced or exhausted while a malicious SOAP message is being parsed.

In summary, the initial security mechanism incorporated within FUSION@ presents the following limitations: a) It can cause a bottleneck during an instance of high service requests and negatively affect the architecture's performance. b) Its strategy can only detect and block a limited number of known attacks, and cannot handle attacks that are more complicated in nature. c) The security mechanism is incapable of adapting to new attack patterns. This limitation prevents the security mechanism from confronting new attacks or fast-paced changes in known attack patterns.

3 AIDeM: Agent-Based Intrusion Detection Mechanism

AIDeM is based on the incorporation of a new security block composed of a set of agents with special capabilities. The new proposed mechanism is based on our previous research in SQL injection attacks [10] [11] which developed a multi-agent SQLMAS architecture. In this way, some resources are reused and the knowledge acquired from previous work is adapted in order to provide an evolution of the mechanism proposed.

Figure 1 presents the AIDeM architecture. As shown in Figure 1, AIDeM is comprised of the Security and Admin agent, both of which were already included

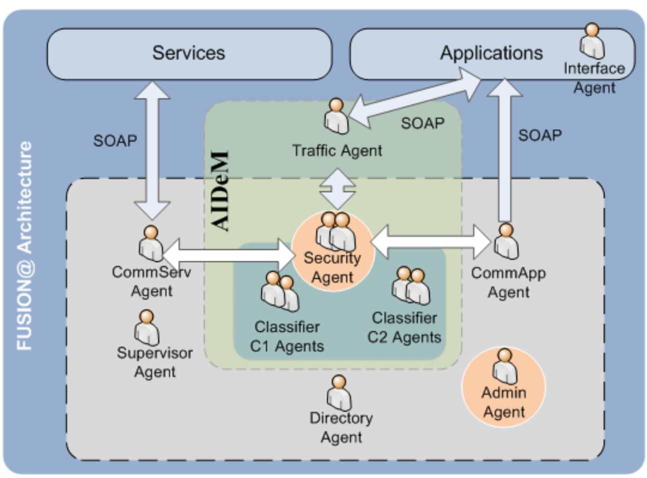


Fig. 1 Representation of AIDeM with the set special agents

in the FUSION@ architecture, as well as the Traffic agent and two others that will function as classifiers: C1 agent and C2 agent. AIDeM was designed as a two phased classification mechanism for classifying SOAP messages, as explained by [10] and [11]. The first phase applies the initial filter for detecting simple attacks without requiring an excessive amount of resources. The second phase involves a more complex process which ends up using a significantly higher amount of resources. This two-phased strategy improves the overall response time of the classification mechanism, facilitating a quick classification of any incoming SOAP messages that were thought to contain significant features during the first phase. The second phase is executed only for those SOAP messages with complex characteristics identified as suspicious during the first phase and requiring a more detailed evaluation. Each of the phases incorporates a CBR-BDI [12] agent with reasoning, learning and adaptation capabilities.

The following section provides a detailed description of the characteristics and tasks related to each of the agents that constitute the AIDeM platform.

- Traffic Agent: This agent has a type of sensor feature that allows it to identify and capture SOAP messages that have been sent from external applications and that request a particular type of service. The agent captures the messages and redirects them to the Security agent for evaluation.
- Security Agent: This agent carries out tasks similar to those assigned with the original FUSION@ security mechanism. The agent is in charge of receiving SOAP message that contain service requests. It performs a quick analysis of the message, and the data obtained are sent to the agent at the first phase of the classification mechanism. With cases that are considered suspicious, the Security agent submits the XML message to a more comprehensive syntactic analysis in order to obtain the necessary data for carrying out the second phase of the classification mechanism. There can be more than one Security agent, depending on the amount of workload.
- Classifier C1 Agent: This is one of the key CBR-BDI agents in the classification process. These agents initiate a classification by incorporating a CBR engine that in turn incorporates a Näives Bayes strategy in the re-use phase. The main goal of this initial phase is to carry out an effective classification, but without requiring an excessive amount of resources. The fields of the case are obtained from the headers of the packages of the HTTP/TCP-IP transport protocol. Table 1 shows the fields taken into consideration to describe the problem.

Table 1 Problem Description First Phase – Classifier C1 Agent

Fields	Type	
IDService	Int	*i*
Subnet mask	String	*m*
SizeMessage	Int	*s*
NTimeRouting	Int	*n*
LengthSOAPAction	Int	*l*
TFMessageSent	Int	*w*

Within the CBR cycle, specifically in the re-use phase, a particular classification strategy is used by applying a Näives Bayes strategy, which gives 3 possible results: legal, malicious and suspicious. Messages that are classified as legal are sent to the corresponding web service for processing. Malicious messages are immediately rejected, while suspicious messages continue through to the classification process executing the second phase of the classification mechanism. There can be more than one Classifier C1 agent depending on the amount of workload.

- Classifier C2 Agent: This CBR-BDI agent completes the classification mechanism. In order to initiate this phase, it is necessary to have previously initiated a syntactic analysis on the SOAP message to extract the required data. Table 2 presents the fields used in describing the problem for the CBR in this layer. Once the data have been extracted from the message, a CBR mechanism is initiated by using a Multilayer Perceptron (MLP) neural network in the reuse phase. There can be more than one Classifier C2 agent, depending on the amount of workload.

Table 2 Problem Description Second Phase – Classifier C2 Agent

Fields	Type	variable
IDService	Int	*i*
MaskSubnet	String	*m*
SizeMessage	Int	*s*
NTimeRouting	Int	*n*
LengthSOAPAction	Int	*l*
MustUnderstandTrue	Boolean	*u*
NumberHeaderBlock	Int	*h*
NElementsBody	Int	*b*
NestingDepthElements	Int	*d*
NXMLTagRepeated	Int	*t*
NLeafNodesBody	Int	*f*
NAttributesDeclared	Int	*a*
CPUTimeParsing	Int	*c*
SizeKbMemoryParser	Int	*k*

- Admin Agent: In addition to the functions already mentioned in the FUSION@ architecture, this agent is responsible for overseeing the correct functioning of the classification process and for coordinating the distribution of tasks.

4 Results and Conclusions

This article has presented AIDeM, a mechanism for classifying incoming requests based on SOAP messages. The mechanism was designed to exploit the distributed capacity of the agents. Additionally, an advanced classification mechanism was designed to filter incoming SOAP messages. The classification mechanism was structured in two phases, each of which includes a special CBR-BDI agent that functions as a classifier. The first phase filters simple attacks without exhausting an excessive amount of resources by applying a CBR engine that incorporates a Naïves Bayes strategy. The second phase, a bit more complex and costly, is in charge of classifying the SOAP messages that were not classified in the first phase.

A small case study was used to evaluate the efficacy of the integration of AIDeM within the FUSION@ architecture. The ALZ-MAS 2.0 multi-agent system [5] was used to carry out the test. It was implemented through FUSION@ and used to construct a tool for dependent environments. In order to evaluate AIDeM, two specific services available in ALZ-MAS 2.0 were selected for external users: *RequestScheduleDoctor()* which is used to consult the agenda of a doctor via Internet and *RequestAppointment()* which is used to request an appointment with a doctor via Internet.

The experiments were carried out in two stages; the first stage was to obtain the test data used for training the classifiers, and the second stage was to evaluate the classification mechanism. In order to obtain the test data in the first stage, the Traffic agent was configured to capture the incoming SOAP messages without redirecting them to the services. In order to send the SOAP messages, 3 points (nodes) were established, from which various requests for selected services were executed. Each of these nodes belonged to a different network, i.e., each node connected to the internet using a different IP and subnetwork mask. In the first stage, each node was configured with 30 requests (SOAP messages) to be sent to each of the 2 selected services. Each node sent a total of 60 requests so that the total number of requests made by the 3 nodes to the 2 services was equal to 180 requests. The 30 requests sent by each node, included legal message and malicious message (incorrectly formed messages). For the second stage of testing, the number of nodes and services was the same as in the first stage, but the number of requests was configured at 15 requests per node. At this stage, once the requests were captured by the Traffic agent, they were sent to AIDeM to be evaluated and classified. A total of 90 requests (legal and malicious) were sent to AIDeM for evaluation. Figure 2 shows the results obtained for the set of SOAP messages evaluated.

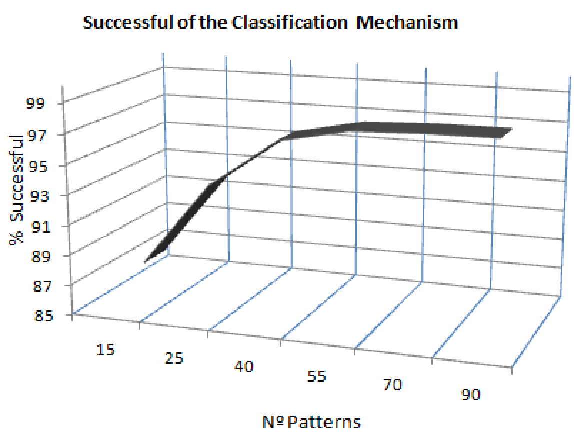


Fig. 2 Effectiveness of the classification mechanism prototype according to the number of patterns

Figure 2 shows the percentage of prediction with regards to the number of patterns (SOAP messages) for the classification mechanism. It is clear that as the number of patterns increases, the success rate of prediction also increases in terms of percentage. This is influenced by the fact that we are working with CBR systems, which depend on a larger amount of data stored in the memory of cases.

Future works are expected to develop the tools for obtaining a complete solution.

Acknowledgements. This development has been partially supported by the Spanish Ministry of Science project TIN2006-14630-C03-03 and The Professional Excellence Program 2006-2010 IFARHU-SENACYT-Panama.

References

1. Negm, W.: Anatomy of a Web Services Attack: A Guide to Threats and Preventative Countermeasures (2004)
2. Nadalin, A., Kaler, C., Monzillo, R., Hallam-Baker, P.: Web Services Security: SOAP Message Security 1.1 (WS-Security 2004) (2006)
3. Bajaj, S., Box, D., Chappell, D., Curbera, F., Daniels, G., Hallam-Baker, P., Hondo, M.: Web Services Policy Framework (WS-Policy) version 1.2 (2006)
4. Anderson, S., Bohren, J., Boubez, T., Chanliau, M., Della, G., Dixon, B.: Web Services Trust Language (WS-Trust) (2004)
5. Corchado, J.M., Tapia, D., Bajo, J.: A Multi-Agent Architecture for Distributed Services and Applications. International Journal of Ambient Computing and Intelligence - IJACI, 15–26 (2009)
6. Aamodt, A., Plaza, E.: Case-based reasoning: foundational issues, methodological variations, and system approaches. AI Communications 7, 39–59 (1994)
7. Cerami, E.: Web Services Essentials Distributed Applications with XML-RPC, SOAP, UDDI & WSDL, 1st edn. O'Reilly & Associates, Sebastopol (2002)
8. Bratman, M.E., Israel, D.J., Pollack, M.E.: Plans and resource-bounded practical reasoning. Computational Intelligence 4, 349–355 (1988)
9. Jennings, N.R., Wooldridge, M.: Applying agent technology. Applied Artificial Intelligence 9, 357–369 (1995)
10. Pinzón, C., De Paz, Y., Bajo, J.: A Multiagent Based Strategy for Detecting Attacks in Databases in a Distributed Mode. In: Corchado, J.M., Rodríguez, S., Llinas, J., Molina, J.M. (eds.) International Symposium on Distributed Computing and Artificial Intelligence - DCAI 2008, vol. 50. Springer, Heidelberg (2008)
11. Bajo, J., Corchado, J.M., Pinzón, C., Paz, Y.D., Pérez-Lancho, B.: SCMAS: A Distributed Hierarchical Multi-Agent Architecture for Blocking Attacks to Databases. International Journal of Innovative Computing, Information and Control (2008)
12. Laza, R., Pavón, R., Corchado, J.M.: A Reasoning Model for CBR_BDI Agents Using an Adaptable Fuzzy Inference System. In: Conejo, R., Urretavizcaya, M., Pérez-de-la-Cruz, J.-L. (eds.) CAEPIA/TTIA 2003. LNCS (LNAI), vol. 3040, pp. 96–106. Springer, Heidelberg (2004)

A Multi-agent System that Searches for Learning Objects in Heterogeneous Repositories

Fernando De la Prieta and Ana Belén Gil

Abstract. This paper presents the BRENHET application, which introduces a new concept in searching for educational resources by using a learning object paradigm that describes these resources. The application is composed of a complete agent-based architecture that implements the concept of federated search. It can search different repositories in parallel, and is based on abstraction layers between the repositories and the search clients.

Keywords: Multi-agent systems, Distributed Computing, e-learning, learning objects, repositories, Simple Query Interface, learning technology standards, web services.

1 Introduction

One of the biggest advances in the field of distance education has been the incorporation of the learning object paradigm[1]. The main advantage derived from the use of these elements is the reusability of the educational resources. These learning objects (LOs) incorporate both a self-contained modular resource and a set of descriptive metadata, which makes it possible to automate any search and sort process.

LOs are typically stored in digital repositories that, being highly heterogeneous, have different storing systems, access to objects, query methods, etc. The problem of heterogeneity in database systems has been the object of much research for quite some time [2] now and, as such, is not particularly critical. However, the fact that the repositories do not contain a high level layer that provides an abstraction between its internal functioning (which should be a black box) and the end users working with the stored data, results in a seemingly insurmountable problem suffered by the majority of repositories. In an attempt to avoid this problem, some

Fernando De la Prieta · Ana Belén Gil
Departamento Informática y Automática
Universidad de Salamanca
Plaza de la Merced s/n, 37008, Salamanca, Spain
e-mail: {fer,abg}@usal.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 355–362.
springerlink.com

repositories have an abstraction layer between their internal and external characteristics, permitting them to better automate and process their learning objects, and allowing for new types of applications referred to as federated search LOs[3].

There are currently various studies underway that look at different ways of encapsulating heterogeneous learning object repositories (LORs). One of the most important, and currently the most used, is the SQI (*Simple Query Interface*) [4] which provides the repositories with an abstraction level between the internal and external heterogeneity that includes a specification with a wide range of possibilities (different service-based communication protocols, synchronous and asynchronous queries, stateless and stateful repositories, etc.).

After an extensive preliminary study, it is apparent that all current repositories use the VSQL[5] query language, which is both simple and powerful, since it is based on XML, and able to adapt perfectly to learning object searches. The semantic specification when searching for learning objects is the LOM (*Learning Object Metadata*) standard, which was formally approved by IEEE in 2002. Although LOM describes the LOs amply and correctly, it does not contain a required specification field. Thus the burden of correctly describing the educational resources is placed squarely on its authors.

This primary purpose of this article is to present the federated search system for LOs in digital repositories, which has been developed specific to this study. Known as BRENHET, the agent-based system uses the SQI interface to perform federated searches in different repositories and sort the retrieved LOs according to the different needs of the application users.

2 Searching for Learning Objects in Repositories

The current increase in resources for online education calls for the efficient management of those services and the elements involved. The greatest efforts within the context of e-learning are now focused on achieving the interoperability of the resources generated. It is within this context that LOs emerged with the aim of reusing educational resources and reducing their production costs [7].

LOs have to be labeled with metadata so that they can be identified, located and used for educational purposes in web-based contexts. The pioneering standardization for LOs was developed by ARIADNE (*Alliance of Remote Instructional Authoring and Distribution Networks for Europe*). The Instructional Management Systems project (IMS) and Advanced Distributed Learning (ADL) initiatives emerged in an effort to develop standards for distributed learning technologies. The former has become the IMS Global Learning Consortium, whose key work has been in vocabularies and metadata for learning objects (IMS Learning Resource Metadata specification). The latter is the home of the SCORM (*Sharable Content Object Reference Model*), a Web-oriented data model for content aggregation focusing on the structure and run-time environment for LOs. SCORM is pretty much accepted as the standard for management of educational content, and its counterpart is a standard for management of educational content metadata [8], Learning Object Metadata (LOM) from IEEE Learning Technologies Standards Committee (LTSC).

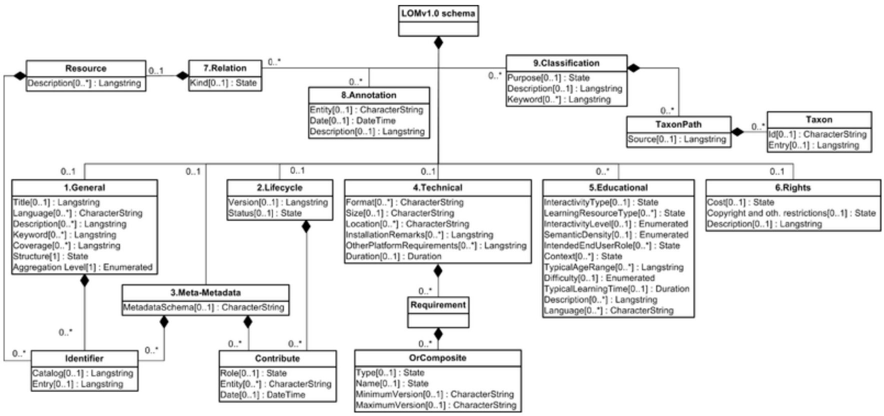


Fig. 1 LOM (Learning Object Metadata)

Both LOM and SCORM were written to be very flexible and extensible, so although learning objects have many variant recommendations, they also have a remarkably large shared foundation.The IEEE LOM standard (IEEE Draft 1484.12.1, 2002), specifies the conceptual schema that defines the structures of the data for instances of LO metadata. The basic schema of LOM is composed of categories and elements as shown in Figure 1, which can describe the resources very well. LOM is able to embed other metadata standards using XML namespaces.

2.1 Learning Objects Repositories

LORs are software systems in the form of digital catalogues that either store educational resources and their metadata, or only the latter. Most LOR belong to educational institutions, as is the case of MERLOT (*Multimedia Educational Resource for Learning and Online Teaching*), CAREO (*Campus Alberta Repository of Education Objects*), or CLOE (*Co-operative Learning Object Exchange*). They generally provide some kind of search interface that makes it possible to retrieve the LOs. Any interaction for retrieving LOs can be done manually or be automated through different software systems. The main standards of interfaces for LORs are CORDRA, IMS-DRI, OKI-OSID or SQI.

- **CORDRA** (*Content Object Repository Discovery and Registration/Resolution Architecture*). This develops an open model for the design and implementation of software systems that retrieve, share and reutilise educational contents through the implementation of federated systems in repositories of educational resources.
- **IMS-DRI** (*IMS Digital Repositories Specification*). IMS Digital Repositories v1.0 are aimed at providing recommendations for the interoperability of the most common repository functions. These recommendations should be implemented through web services so that they have a common interface. On the

broadest level, this specification defines digital repositories as being any collection of resources that are accessible via a network without prior knowledge of the structure of the collection.

- **OKI OSID** (*Open Knowledge Initiative Open Service Interface Definition*). The OKI specification allows sustainable interoperability and integration by defining standards based on Service Oriented Architecture (SOA). OSID is simply the contract software and thus is compatible with most other technologies and specifications, such as SOAP and WSDL.
- **SQI** (*Simple Query Interface*), defined in November 2005 by CEN (*European Committee of Standardization*), is defined by three APIs: Learning Objects Interoperability Framework, Authenticationand Session Management, and Simple Query Interface Specification, which provides a detailed description of the functionalities that each repository must meet in order to be compatible with the others, and describes the names of the methods that must be implemented along with the parameters received and returned during the search process.

It is thus imperative to be able to rely on an efficient and interoperable search system that can guarantee the recovery of learning objects from the repositories. For this reason we have designed an SQI-based metasearch system that can perform a federated search in distributed repositories.

3 Overview of BRENHET

We have built a LO federated search system for digital repositories that is completely functional and accessible by any user. In addition to searching for LOs, the system catalogues objects that are returned by the LORs, thus facilitating the final selection made by the end users. The final sorting is based on two variables. The first estimates the quality of the objects according to different parameters such as size, included key characteristics, etc. The second variable looks at the end user activity to classify the recovered objects. This provides the classification process with techniques that come directly from a collaborative recommendation of network information (Collaborative-Social-filtering systems[6]).

Fig. 2 BRENHET. Federated search system.

Two search interfaces are provided. One is simple and quick, allowing the user to input a series of key words or an exact phrase that should contain the metadata of both. There is also a more advanced and powerful search that provides the user with a greater number of search options.

As part of its administrative functionality, the BRENHET system implements a new repository management system that provides the application with an inclusion protocol for new repositories. This protocol incorporates an automated validation system of repositories prior to their inclusion, that ensures the application and the search system are autonomous and reliable.

The functionality of the application is enhanced and its use greatly facilitated by the implementation of the following additional functionalities: user management, configuration management, search history, and a complete system of statistics based on statistical data gathered in each query from each repository.

3.1 BRENHET Search Architecture

In order to develop the BRENHET application, it was necessary to design and implement an agent-based architecture specifically focused on resolving the problem of the federated search in repositories that store learning object through the use of a SQI query interface.

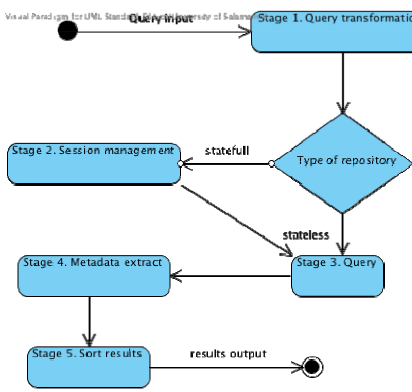


Fig. 3 Sequence of task in federated search

As shown in the figure above, the query and the results output processes are similar for each query and include the following phases:

1. **Query transformation.** During the development of this phase, the query is transformed into a more common language that the user inputs in the application. It is a VSQL language that, as previously mentioned, is the query language for repositories that implement the SQI interface.
2. **Session management.** This phase only needs to be executed in statefull repositories. The identifier is obtained through a query using web services.

3. **Query.** Once the session identifier, whether static or dynamic, is available, the next step that must becarried out is the actual query. A synchronous query is implemented so that the results are obtained as an answer to the web services query.
4. **Metadata extract.** Once the results have been obtained, the information contained in the metadata for each learning object is extracted.
5. **Sortresults.** Once all the learning objects are obtained, the results are ordered so that they can be presented to the user more properly. This selection process must consider the use made of the learning objects by the end users, and the quality of the objects themselves.

Phase 1 transforms natural language into VSQL language. Within the context of this type of query, the transformation phase is performed in a direct manner, which can be fully appreciated in Figure 4.

dog "learnin-gobject"	➔	"dog" or "lear-ningobject"	➔	<simpleQuery> <term>dog</term> <term>learningobject</term> </simpleQuery>

Fig. 4 VSQL transformation

Phases 2 and 3 of the algorithm constitute the majority of the time used in the query, since they make a series of requests to the LORs using web services. Phase 4 processes the results so that they can be more coherently ordered in the subsequent phase. This is possibly the most important phase of the algorithm since it considers various premises that can be divided into two primary groups:

1. **Importance for the user.** As users navigate the application, data is gathered from the use of LOs and later used during the ordering phase.
2. **Quality of the LOs.** The quality of the actual learning object is considered (key elements, number of elements, relationship with other objects, etc.).

During the algorithm design phase, the response time of the repositories was too high, just as with the results of initial study. One solution to this problem was to deploy the previously described phases 2, 3 and 4 in parallel, and to implement a timeout system. This reduced the response time to the end user and resulted in a completely functional application.

The BRENHET search system architecture is composed of three basic blocks: Interface, Search system, and Communication protocols. These blocks constitute the foundation of the system's functionality. The interface is primarily used for a bidirectional communication between the users and the application. The search system is implemented through the use of agents that are responsible for security, communication and the taskflow in BRENHET. This block is the core of the application and integrates two types of agents that are each assigned to different types of tasks according to each one's primary objective. The first group is composed of deliberative BDI agents, which are responsible for keeping track of and coordinating the areas, can modify their behavior according to the knowledge acquired in order to find the best solution. These agents base their decision on the knowledge acquired and on a series of statistical data that is gathered during each interaction with the repositories that contain LOs. This provides the architecture with great flexibility, even for incorporating new agents in the future, which helps improve the overall time and functioning of the global search process.

The following list describes the predefined agents that provide the basic functionality of the BRENHET architecture:

- **Comunication Agent.** Responsible for all the communication between the different repositories and the application. Manages all of the requests and responses simultaneously. These agentsuse Web services to communicate, and send and receive messages in XML format.
- **Security Agent.** Responsible for verifying the XML message sreceived and confirming that they are correct, and that the LOs are of a good enough quality to be reused.
- **Statistical Agent.** Responsible for taking statistical data from each exchange of communication with the repositories, and presenting the data to the other agents so that they can make the best decisions at any given time.
- **Sort Agent.** Responsible for sorting the results according to the statistical data on the use of the learning objects made by the end users.
- **Supervisor Agent.** This agent´s task is to supervise and ensure that the other agents are functioning correctly and that the application provides the desired functionality.
- **Optimize Agent.** Responsible for optimizing system performance with respository queries.

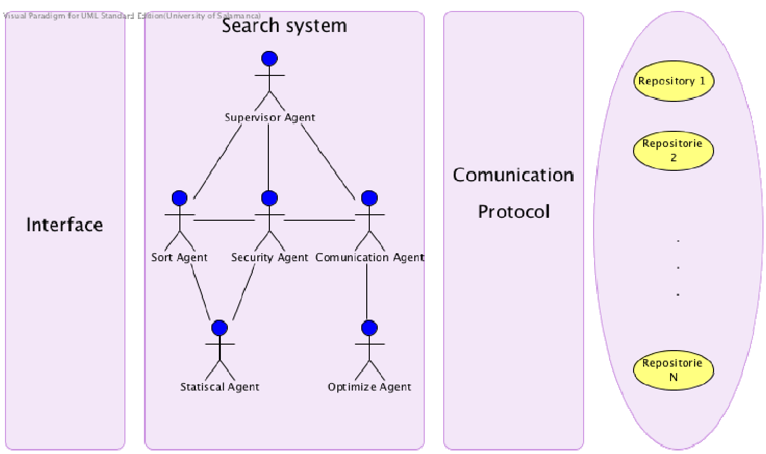


Fig. 5 BRENHET Architecture

The second group of agents is based on a reactive agent platform that primarily aims to improve the application's overall response time. These agents must take into account a series of predefined temporal restrictions so that they can be capable of autonomously controlling the overall quality and reliability of the application.

Finally, the communication protocol permits the applications, services and sensors to be continually interconnected with the agent-based search system. This protocol is based on the SOAP standard and allows the exchange of messages between applications, services and repositories.

4 Results and Conclusions

This paper has provided a brief glimpse into the BRENHET System, which resolves the problem of heterogeneity in repositories that contain LOs, and the difficulty in processing the repositories that arises from the inconsistencies within the LOM specification. BRENHET is a system that searches for LOs with a federated search. It has an agent-based architecture that uses an SQI interface to access Web services for heterogeneous repositories that store LOs, to select the required objects, and sort them according to user preferences.

Taking into account the results obtained from the construction of the BRENHET application, it is possible to conclude that the agent-based architecture is ideal for resolving the problem of federated searches in heterogeneous repositories, due to the adaptation and learning facilities of the actual agents.

References

1. Lujara, S.K., Kissaka, M.M., Bhalaluseca, E.P., Trojer, L.: Learning Objects: A new paradigm for e-learning resource development for secondary schools in Tanzania. World Academy or Sciencie, Engineering and Technology, 102–106 (2007)
2. Abelló, A., Oliva, M., Samos, J., Saltor, F.: Information System Architecture for Data Warehousing from a Federation. In: Int. Workshop on Engineering Federated Information Systems, EFIS 2000 (2000)
3. Gil-González, A.-B., García-Peñalvo, F.: Multiagent system for learning objects recovery with context attributes. Int. J. Computer Applications in Technology 33(4), 320–326 (2008)
4. Campi, A., Ceri, S., Dolog, P., Duval, E., Guinea, S., Houben, G., Massart, D., Nilsson, M., Ternier, S., Xuan, Z.: ARIADNE foundation for the European Knowledge Pool. ProLearning Query Language Definition, V.0.7, `http://ariadne.cs.kuleuven.be/lomi/index.php/QueryLanguages_stable`
5. Simon, B., Massart, D., Van Assche, F., Ternier, S., Duval, E., Brantner, S., Olmedilla, D., Miklos, Z.: A Simple Query Interface for Interoperable Learning Repositories. In: 1st Workshop on Interoperability of Web-based Educational Systems, Chiba. Japan (2005)
6. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. 22(1), 5–53 (2004)
7. Hatala, M., Richards, G., Eap, T., Willms, J.: The interoperability of learning object repositories and services: standards, implementations and lessons learned. In: Proceedings of the 13th international World Wide Web Conference on Alternate Track Papers &Amp; Posters, WWW Alt. 2004, May 19 - 21, pp. 19–27. ACM, New York (2004)
8. Ogbuji, U.: Think XML: Learning Object metadata, XML Knowledge management flourishes in learning technology initiatives (2003), `http://www.ibm.com/developerworks/xml/library/x-think21.html`

Application of a Modeling Standard Language on the Definition of Agent Oriented Development Processes

Alma M. Gómez-Rodríguez and Juan C. González-Moreno

Abstract. The study of development processes is a key issue regarding quality of final products and reutilization of process portions or fragments. In this work, the process of a particular methodology for Agent Oriented Development is defined using a standard notation, in particular SPEM 2.0 . This case study shows the suitability of process definition as a starting point for reuse of the development process in the Agent Oriented Software Engineering field. Moreover, the results obtained will permit to design a customizable CASE tool that can fit in any development process previously defined.

1 Introduction

The software quality assurance considers that the development process is very important because of the direct relation between process quality and final product quality. In particular, in Agent Oriented Software Engineering (AOSE) many methodologies and their associated processes of development have been proposed in the latest years [5, 2, 7, 15]. All of them introduce all the conceptual abstractions that must be taken into account in any MultiAgent System (MAS) development.

Nevertheless, in AOSE field few attention has been paid, until the moment to process definition. All this encourages the necessity of obtaining models of the processes for the development of MAS, that define its structural and behavioral issues, as it has been introduced in FOSE-MAS [21]. For defining MAS based processes, the *Foundation for Intelligent Physical Agents* (FIPA) through its *Methodology Technical Committee* [4] has suggested the use of Software Process Engineering

Alma M. Gómez-Rodríguez
Departamento de Informática, University of Vigo. Ed. Politécnico, Campus As Lagoas, Ourense E-32004, Spain
e-mail: `alma@uvigo.es`

Juan C. González-Moreno
Departamento de Informática, University of Vigo. Ed. Politécnico, Campus As Lagoas, Ourense E-32004, Spain
e-mail: `jcmoreno@uvigo.es`

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 363–370.
springerlink.com

Metamodel (SPEM), whose current version is 2.0 [17]. SPEM is a standard notation defined by the Object Management Group (OMG) to model software development processes. As the SPEM standard is the one recommended by FIPA and is well documented with available information, it was the one selected for this proposal. Moreover, previous works in the field [20, 10, 8, 11] have used SPEM as the standard for Agent Oriented Process modeling.

Following these works, as well as other lines proposed by the FIPA Design Process Documentation and Fragmentation Working Group, this paper addresses the definition of a well known development process in the AOSE field, the development process proposed by INGENIAS methodology [12, 14, 18] which is based on Rational Unified Process (RUP)[19] and Unified Development Process(UDP).

The rest of the paper is organized as follows. After this Introduction, in section 2 a general overview of INGENIAS Process is presented. Next section 3 addresses the definition of Inception Phase of INGENIAS Process, that has been taken as significative case study. Finally, the paper ends with the Conclusions and Future Work section.

2 Global Overview of the INGENIAS Process

2.1 Introduction

Initially a general overview of the process at a high level of abstraction must be provided. For achieving this goal, this section will deal with issues such as: the global description of the process and its phases or the key concepts defined during the development.

The process taken as case study for this paper is INGENIAS-UDP. INGENIAS methodology covers the analysis and design of MAS and it is intended for general use; that is, with no restrictions on application domains. The methodology is supported by the INGENIAS Development Kit (IDK), which contains a graphical editor for MAS specifications. Besides, the INGENIAS Agent Framework (IAF) [13] integrated in the IDK has been proposed for enabling a full model-driven development and transforming automatically specifications into code in the Java Agent DEvelopment Framework.

INGENIAS tries to follow a Model Driven Development(MDD) [1], so it is based on the definition of a set of meta-models that describe the elements that form a MAS from several viewpoints. The specification of a MAS is structured in five viewpoints [16]:

1. The definition, control and management of each agent mental state.
2. The agent's interactions.
3. The MAS organization.
4. The environment.
5. The tasks and goals assigned to each agent.

Detailed references of the methodology from their authors can be found in [12], [14, 16].

2.2 Description of the Process

Like UDP, INGENIAS methodology distributes the tasks of Analysis and Design in three consecutive phases: Inception, Elaboration and Construction, with several iterations (where iteration means a complete cycle of development, which includes the performance of some analysis, design, implementation and proofs tasks). The sequence of iterations leads to the procurement of the final system [12].

Fig. 1 Lifecycle for INGENIAS Methodology

A detailed description of the Unified Development Process (UDP) of INGENIAS methodology can be seen in [12, 14, 16]. Nevertheless a general view of the process is introduced in Figure 1. The three development phases are considered to have two different types of workflow: Analysis and Design. The methodology pays few attention, compared to RUP, to Implementation and Test workflows, because it provides some tools which automatically generate code, in parallel with system's specification. Attending this facility, these workflows are considered not to be modeled as fundamental part of the process.

Figure 1 is a SPEM activity diagram where only phases are considered. In SPEM phases are considered as a special kind of activity which represents a significant period in a project, ending with major management checkpoint, milestone, or set of Deliverables [17].

2.3 INGENIAS Metamodel

INGENIAS is based on the concept of metamodel. A metamodel defines the primitives and the syntactic and semantic properties of a model. As previously stated, INGENIAS provides five meta-models that constitute the five different views of the system. There are some key elements in the metamodel which relate the different views of the system. The particular elements of each metamodel as well as their relations and definitions can be found in [12, 14, 16]. Metamodels are the key issue in MAS development, according to INGENIAS, because they constitute the basis of MAS specification. Nevertheless, these models and the activities done to obtain them must integrate in a Software Engineering Process. This integration will be addressed in this section and the next ones.

After Introduction, each of the phases of the process will be described with a higher level of detail. In the case of INGENIAS-UDP, Inception, Elaboration and Construction must be detailed. For sake of brevity, in the paper only the Inception phase will be addressed.

3 Inception Phase

INGENIAS considers that the development initiates from the document describing the problem. This means that this document will be considered as an initial input of the process. Using it as input, the Inception phase introduces several activities, described in Figure 2.

Regarding the Analysis workflow these activities must be done in this phase:

- Generate Use Cases
- Generate the Environment Model
- Initiate the architecture using the Organization Model

In what respects to Design only the construction of a rapid prototype must be addressed.

All these activities and the associated tasks are shown in Figure 2. From this figure, the different tasks proposed by INGENIAS for Inception and the workproducts produced, can be identified. Moreover, the roles responsible of each task as well as the kind of responsibility they assume are also shown.

Figure 2 is a SPEM Activity Diagram which shows some details on the activities and tasks to be done. It also shows the workproducts obtained as outputs or needed as inputs for each of the activities proposed, as well as the roles responsible for each activity. Roles and activities are further detailed in next subsections.

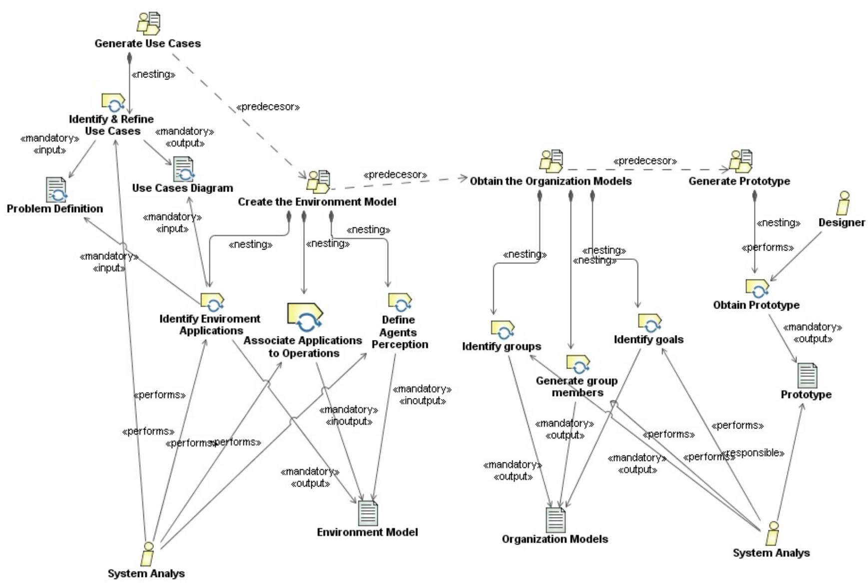


Fig. 2 Detailed tasks of Inception activities

3.1 Process Roles

INGENIAS methodology makes no explicit reference to the roles implied in the development. Nevertheless, regarding the focus of this paper on INGENIAS process and its adaptation to UDP, we considered that two roles are implied in the process: the System Analyst and the Designer.

The System Analyst is responsible or performs the most part of the activities proposed in this phase. In particular, he will:

- Identify the Use Cases and construct and refine the Use Cases Diagram. From the initial description of the problem to solve, the analyst must obtain the use cases that will guide the creation of the Interaction Model.
- Define the Environment Model, showing the interaction of the system with its environment. This will imply to: identify applications (in INGENIAS, all the software and hardware that interact with the system and can't be designed as agent will be considered an application); associate operations to particular applications and define agents perception on applications.
- Obtain the Architectural view of the System using the Organization Model. This means to generate a structural definition of the system by identifying groups in the organization, generating group members and identifying goals.

The second role identified in this phase has been the *Designer*. He must be responsible of generating the prototypes. According to INGENIAS literature, this will be done using a rapid application development tool such as ZEUS, Agent Tool or others.

3.2 Activity Details

This section details the activities previously outlined for Inception Phase. Each of the subsections will consider a particular activity.

3.2.1 Generate Use Cases

The generation and refining of Use Cases has been identified as a unique task. The goal of this task will be to identify the intended functioning of the system. Knowing the functionalities the system must provide, will allow to identify collaborators and initiators in interactions and also to discover the nature of such interactions that will affect the type of control applied to the agent: planning, cooperation, contract-net or competition.

3.2.2 Generate the Environment Model

The Environment Model tries to show the elements that constitute the environment of the system, and in consequence, what the agents have to perceive. The elements defined in this model are of three basic kinds: agents, resources and applications.

Obtaining the Environment Model of the system to construct comprises a set of tasks, that must be done by System Analyst, and are explained next:

- *Identify Environment Applications.* All the software and hardware that interact with the system and that can not be designed following an agent oriented approach will be considered an application.
- *Associate Applications and Operations.* In this task, operations are associated to the applications defined by requirements. These operations have a signature, a precondition and a postcondition. The identification of operations is an conventional engineering task.
- *Define Agents Perception.*The main aim of this task is to define agents perception on environment applications, at this moment of process it is enough to relate agents and applications.

3.2.3 Initiate the Architecture

One of the key activities in Inception Phase is to start the definition of the system architecture. This is done by constructing the Organization Model, which reflects mainly the system's workflows.

The basic tasks related with the procurement of Organization Model in Inception activity are shown next. These activities try to obtain an organizational view of the system, attending its structural, functional and social aspects and are done by System Analyst role.

- *Identify groups.* The groups in the system must be identified. In this way the participants in a particular workflow will be organized.
- *Generate group members.* Members (agents, roles, resources and applications) are assigned to groups creating the corresponding relationships. If needed, the groups can be decomposed in order to reduce complexity.
- *Identify groups.* The organization has a set of goals that must justify collaboration between agents. The goals identified in this task will after be assigned to individual agents or roles in the Task and Goals Model.

3.2.4 Construction of a Prototype

The generation of a prototype is a unique and simple task. As said previously, it will be generated using a RAD tool.

4 Conclusions and Further Work

This work has addressed the definition of the INGENIAS process (in particular its Inception phase), using SPEM 2.0 standard as notation.

The definition of processes is very important in Software Engineering as it makes easier their study and improvement. Besides, defining a process is the best way of documenting it. Documentation constitutes a key issue for understandability and usability.

Although some works have addressed process comparison in AOSE [3, 6], if all the processes were defined in the same or similar way and with the same notation, they can be compared much more easier. Afterwards, and considering comparison conclusions, the most suitable process for a particular development can be chosen on the members of the team (or vice versa). Moreover the process can be dynamically changed during the development, if it is correctly defined.

Other advantage of having a formal defined process is related with the tools supporting the development. At the moment, each particular methodology and process provides its tools for development. Once we have the development processes completely defined, this definition can be used as input for a generic tool, that will implement the process taken as input. This tool will be customized by the process definition and will guide the user in the different activities s/he has to do, the workproducts to obtain, etc. An initial step in this approach has been made in [9], where the definition of the system is obtained in a XML document, that can be used as input in the tool proposed.

In the future, we will go on working in process formalization, trying in this way to provide the scientific community with standard definitions of other INGENIAS processes.

Acknowledgements. This work has been supported by the project *Novos entornos colaborativos para o ensino* supported by Xunta de Galicia with grant 08SIN009305PR.

References

1. Atkinson, C., Kuhne, T.: Model-driven development: a metamodeling foundation. IEEE Software 20(5), 36–41 (2003)
2. Bernon, C., Cossentino, M., Pavón, J.: Agent-oriented software engineering. Knowl. Eng. Rev. 20(2), 99–116 (2005)
3. Cernuzzi, L., Cossentino, M., Zambonelli, F.: Process models for agent-based development. Engineering Applications of Artificial Intelligence 18(2), 205–222 (2005)
4. Cossentino, M., Garro, A.: Activity of the FIPA Methodology Technical Committee. Technical report, Consiglio Nazionale delle Ricerche (2005)
5. Cuesta, P., Gómez-Rodríguez, A., González-Moreno, J.C., Rodríguez, F.J.: The MESMA methodology for agent-oriented software engineering. In: Proc. of First International Workshop on Practical Applications of Agents and Multiagent Systems (IWPAAMS 2002), pp. 87–98 (2002)
6. DeLoach, S.A., Padgham, L., Perini, A., Susi, A., Thangarajah, T.: Using three aose toolkits to develop a sample design. Int. J. Agent-Oriented Softw. Eng. 3(4), 416–476 (2009)
7. Bergenti, F., Gleizes, M.P.: Methodologies And Software Engineering For Agent Systems: The Agent-oriented Software Engineering Handbook. Springer, Heidelberg (2004)
8. García-Magariño, I., Gómez-Rodríguez, A.M., Gómez-Sanz, J., González-Moreno, J.C.: Ingenias-SCRUM development process for multi-agent development. In: Int. Symposium on Distributed Computing and Artificial Intelligence (DCAI 2008). Advances in Software Computing (2008)

9. García-Magariño, I., Gómez-Rodríguez, A.M., González-Moreno, J.C.: Modeling INGENIAS development process using EMF. In: 6th International Workshop on Practical Applications on Agents and Multi-agent Systems, IWPAAMS 2007, pp. 369–378 (2007) (in Spanish)
10. García-Magariño, I., Gómez-Rodríguez, A., González-Moreno, J.C.: Definition of process models for agent-based development. In: Luck, M., Gomez-Sanz, J.J. (eds.) AOSE 2008. LNCS, vol. 5386, pp. 60–73. Springer, Heidelberg (2009)
11. Gómez-Rodríguez, A.M., González-Moreno, J.C.: Methodology vs. Development Process: A Case Study for AOSE. In: Omatu, S., Rocha, M.P., Bravo, J., Fernández, F., Corchado, E., Bustillo, A., Corchado, J.M. (eds.) IWANN 2009. LNCS, vol. 5518, pp. 29–36. Springer, Heidelberg (2009)
12. Gómez-Sanz, J.: Modelado de Sistemas Multi-agente. PhD thesis, Universidad Complutense de Madrid. Facultad de Informática (2002)
13. Gómez-Sanz, J.: Ingenias Agent Framework. Development Guide V. 1.0. Technical report, Universidad Complutense de Madrid (2008)
14. Gómez-Sanz, J., Pavón, J.: Meta-modelling in agent oriented software engineering. In: Garijo, F.J., Riquelme, J.-C., Toro, M. (eds.) IBERAMIA 2002. LNCS (LNAI), vol. 2527, pp. 606–615. Springer, Heidelberg (2002)
15. Henderson-Sellers, B., Giorgini, P.: Agent-oriented methodologies. Idea Group Pub., Hershey (2005)
16. Grupo de Investigación en Agentes Software: Ingeniería y Aplicaciones. INGENIAS Section (2009), http://grasia.fdi.ucm.es/main/?q=es/node/61
17. O.M.G., Software Process Engineering Metamodel Specification. Version 2.0, formal/2008-04-01 (2008), http://www.omg.org/
18. Pavón, J., Gómez-Sanz, J.: Agent Oriented Software Engineering with INGENIAS. In: Mařík, V., Müller, J.P., Pěchouček, M. (eds.) CEEMAS 2003. LNCS (LNAI), vol. 2691, pp. 394–403. Springer, Heidelberg (2003)
19. Rational Software, Rational Unified Process: White Paper (1998)
20. Seidita, V., Cossentino, M., Gaglio, S.: Using and extending the SPEM specifications to represent agent oriented methodologies. In: Luck, M., Gomez-Sanz, J.J. (eds.) AOSE 2008. LNCS, vol. 5386, pp. 46–59. Springer, Heidelberg (2009)
21. Weyns, D.: The Future of Software Engineering and Multiagent Systems. Held at AAMAS 2008 (2008)

Variable Length-Based Genetic Representation to Automatically Evolve Wrappers

David F. Barrero, Antonio González, María D. R-Moreno, and David Camacho

Abstract. The Web has been the star service on the Internet, however the outsized information available and its decentralized nature has originated an intrinsic difficulty to locate, extract and compose information. An automatic approach is required to handle with this huge amount of data. In this paper we present a machine learning algorithm based on Genetic Algorithms which generates a set of complex wrappers, able to extract information from the Web. The paper presents the experimental evaluation of these wrappers over a set of basic data sets.

1 Introduction

The Web has been (and is) a success. However the explosion of contents originated in the late 90's, combined with the lack of centralized organization, has made the Web something like the "Wild West". Information is generated and stored without well defined policies and mechanisms to locate and extract it. This scenario makes locating and extracting information a hard task, thus automated solutions have to be created. In this context of unstructured information is where wrappers take an important role. Wrappers are specialized programs that automatically extract data from documents and convert the information stored into a structured format (Camacho et al., 2008). In this paper we propose the use wrappers that extract information using evolved paragraph regular expressions (Friedl, 2002), or simply *regex*.

David F. Barrero · María D. R-Moreno
Departamento de Automática. Universidad de Alcalá,
Ctra Madrid-Barcelona, Km. 33,6. 28871 Alcalá de Henares Madrid, Spain
e-mail: {mdolores,david}@aut.uah.es

Antonio González · David Camacho
Departamento de Informática. Universidad Autónoma de Madrid,
C/Francisco Toms y Valiente, n 11, 28049 Madrid, Spain
e-mail: {antonio.gonzalez,david.camacho}@uam.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 371–378.
springerlink.com

From a practical perspective, a regex is a string that defines a pattern, and thus it can be used to perform tasks such as pattern identification or string extraction. Regex have been widely used by sysadmins and programmers for decades to process log files, validate user inputs or extract data from the Web. Regular expressions are a powerful tool, however they have a pronunced learning curve. A regular expression, in Formal Language Theory, is a characterization of a regular language (Brookshear, 1989) and thus a regex fully describes a DFA or NFA.

The problem of automata learning from examples is a well known problem, first described by (Gold, 1967), and is generally known as *language induction*. It is usually recognized that Evidence Driven State Merging (EDSM) (Lang, 1998) algorithms have good performance. However, in some contexts, using a regex is more natural, and its syntax can be used as a more convenient representation of the underlying automata (Petry et al., 1994).

We aim to extract information with a wrapper able to learn a regex from a set of examples. Our approach uses Genetic Algorithms (GA) (Holland, 1992; Goldberg, 1989). The algorithm is feed with two sets of strings: one containing samples of the pattern that are meant to be extracted and another one with strings that should not be extracted. The algorithm has been integrated within a information extraction and integration tool called Searchy (Barrero et al., 2005). We have previously used a GA with an islands model (Barrero et al., 2009) to generate regex, which lead to complex and domain dependent setups. In this paper we aim to simplify the wrapper through a Variable-Length Genetic Algorithm (VLGA) (Ramsey et al., 1998).

This article is structured as follows. The evolutionary regex wrapper is presented in the second section while some experiments carried out by the regex wrapper are shown in section 3. Then some conclusions and future work are presented.

2 Evolutionary Regular Expression Wrapper

Wrappers allow to isolate the problem of Information Extraction in complex and distributed systems. It encapsulates the pattern, or patterns, to be handle and any kind of method, algorithm, or programming language can be used. They may use any algorithm able to extract information, for instance, GAs.

GA is a type of algorithm that is inspired in the biological process of evolution to address a wide range of Artificial Intelligence (AI) problems. Natural evolution involves a population of individuals characterized by their genetic code. These individuals are under a selective pressure where those ones that are better adapted to the environment have more chances to survive. This selective pressure, in conjunction with changes to the genetic code made by mutation and sexual reproduction, forces the evolution of the population.

A GA have to set, at least, a population of potential solutions characterized by an artificial chromosome, a mechanism to measure the quality of the solution that the chromosome represents, and a set of genetic operators. The canonical GA defines a binary chromosome of fixed length (Holland, 1992). When the size of the solution is not known, using VLGS might be more suitable Hutt and Warwick (2007).

The VLGA implemented in the proposed wrapper uses a binary chromosome divided in several genes of fixed length. Each gene codes a symbol from an alphabet composed by a set of valid regular expressions constructions. It is worth to point out that the alphabet is not composed by single characters but by any valid regex. These simple regular expressions are the building blocks of all the evolved regex and cannot be divided, thus, we will call them atomic regex. The position (or *locus*) of a gene determines the position of the atomic regex. Gene in position *i* is mapped in the chromosome to regex transformation as an atomic regex in the position *i*.

An example may clarify the mapping process. Figure 1 represents how the regex *cats*+ and c*ts*? can be coded in the GA with a gene size of 5 bits. Here, the first letter of *cats*+, *c*, is coded with the first gene of the chromosome, which is the binary string *11100*. In this case *c* is a single letter, but it may be a more complex regex, depending on how the alphabet had been constructed. Following with the example, the second letter of *cats*+ is *a*, it is coded in the second gene of the chromosome with the string *00000*, and so on.

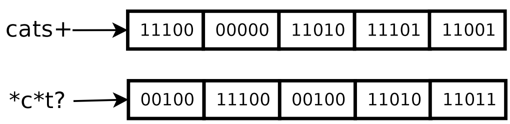


Fig. 1 Example of chromosome encoding

We use classical mutation and crossover operators. Since the codification relies in a binary representation, the mutation operator is the inverse operation meanwhile the recombination is performed with a cut and splice crossover. Given two chromosomes, this crossover selects a random point in each chromosome and use it to divide it in two parts, then the parts are interchanged. Obviously, the resulting chromosomes will likely be of different lengths. Cut and splice is the genetic operator that generates chromosomes length diversity since mutation does not modify the chromosome length. Selection is done by means of a tournament with a size described in section 3. Initial population is generated randomly with chromosome lengths uniformly distributed between two values.

The fitness function is a key subject in the construction of a GA. In our case, for each positive example, the proportion of extracted characters is calculated. Then the fitness is calculated subtracting the average proportion of false positives in the negative example set to the average of characters correctly extracted. In this way, the maximum fitness that a chromosome can achieve is one. It happens when the regex has extracted correctly all the elements of positive examples while no element of the negative examples has been matched. An individual with a fitness value of one is called *ideal individual*.

The implementation of the evolutionary regex was done using a wrapper platform (Barrero et al., 2005). This platform provides a set mapping rules that ease the process of transform unstructured data into RDF. The wrapper first generates a valid regex for each term executing the described VLGA. Once a suitable regex is generated, the wrapper can begin to extract records from any text file accessible thought HTTP or FTP. The extraction capabilities of the described wrapper are evaluated in the next section.

3 Evaluation

Evaluation has been divided in three phases: a first one where the GA parameters are selected; a second phase that generates the regex and a third one where the wrapper uses the evolved regex to extract data. Experiments have used two datasets to evolve two regex, one able to extract emails and another one able to extract US phone numbers. A common issue when working with GAs is parameter tunning, which is the concern of the first experimental evaluation. GAs are characterized by a large number of parameters, and, althought they are quite robust, their performance might be affected by the parameter setting. So, some initial experiments were carried out to acquire knowledge about the behavior of the regex evolution and select the GA parameters to use within the wrapper. Parameters are related in a complex non-linear way. The experiment measured, for each mayor parameter, the mean fitness in a range values of that parameter. Then, the value with higher fitness is selected. The parameters that have been set in this way are the mutation probability, population size, tournament size and elitism size.

Experiments showed that, despite the differences between the phone and email records, both have similar behaviors. In this way it is possible to extrapolate the experimental results and thus to use the same GA parameters. Setup experiments showed that a good performance is achieved with a mutation probability of 0.002 (see Fig. 2) and a tournament size of 2 individuals. A population composed by fifty individuals is a good trade-off between computational resources and convergence speed. It has been randomly generated with a chromosome size that ranges from

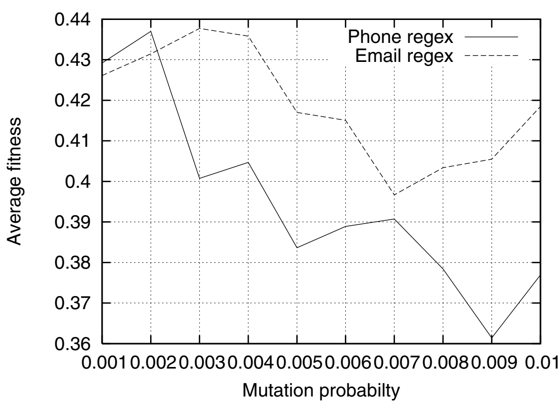


Fig. 2 Fitness versus mutation probability

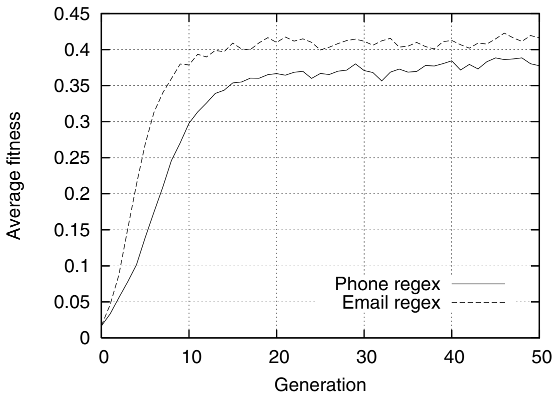


Fig. 3 Evolution of mean fitness

3 to 24 bits. Due to the stochastic nature of GAs, all experiments were run one hundred times and the measures have been averaged. Once the main parameters have been set, they can be used by the evolutionary regex wrapper. Figure 3 depicts the evolution of email and phone regex average fitness. It can be seen that the behavior of both fitness are similar, however email regex fitness converges faster, achieving a fitness value slightly higher than phone regex. The percentage of successful runs is 94% in case of phone regex, and it is increased to the 97% for email regex. Some evolved regex and their associated fitness can be seen in Table 1.

The dynamics of the chromosome length can be observed in the Fig. 4, where the average chromosome length is depicted. It is clear that there is a convergence of the chromosome length and thus chromosome bloating (Chu and Rowe, 2008) is not present. This fact can be explained considering the lack of non-coding and overlapping regions in the chromosome coding, i.e., an ideal individual can only be achieved thought a certain chromosome length, so longer or shorter individuals achieve lower fitness and therefore they are discarded by the action of selective pressure. High fit individuals can be achieved only by chromosomes of a determined size, even if it is not explicit in the fitness function. This correlation between fitness and chromosome size is confirmed by a comparison between Figs. 3 and 4. Email fitness has a faster convergence and it also reaches its minimum chromosome length earlier.

Fig. 4 shows another interesting behavior. When the evolution begins, the average chromosome length tends to decrease until it reach a minimum, then it begins to increase to converge into a fixed value. It can be explained by the relationship between chromosome length and its fitness. Long chromosomes generates long regex, and thus the regex extracts more restricted patterns, decreasing the fitness associated with the regex. In early generations, individuals have not suffered evolution and thus its genetic code has a strong random nature. As a result long chromosomes are discarded in early stages of the evolutive process. Once the population is composed by individuals with basic regex, the recombination increases the

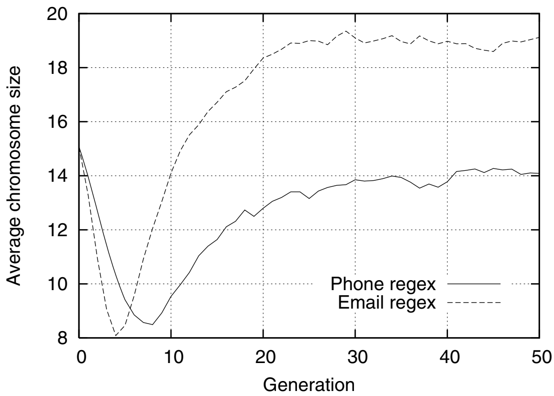


Fig. 4 Mean chromosome length

Table 1 Evolved regular expressions

Evolved regex (email)	Fitness	Evolved regex (phone)	Fitness
\w+\.-com\d+@	0	\w+	0
\w+\.	0.26	\(\d+\)	0.33
\w+@\w+\.	0.78	\(\d+\)\d+	0.58
\w+@\w+\.com	1	\(\d+\)\d+-\d+	1

complexity of individuals merging good chromosomes chunks, increasing the average chromosome length. The observed behavior is aligned with the literature about VLGA (Burke et al., 1998; Hutt and Warwick, 2007).

When the GA has finished, the wrapper selects an ideal individual and use it to extract data. In this third experimental phase extraction capabilities are evaluated by means of the precision, recall and F-measure. Experiments use a dataset composed by six documents containing phone numbers and emails. Table 2 shows basic information about the dataset and its records. Examples have been divided in a trainning set and a testing set. Documents one, two and three are composed by the testing set. The rest of the documents are web pages retrieved from the Web without further manipulation. It should be noticed that an extracted string is computed as a correct extraction if and only if it matches exactly the record, otherwise it has been computed as a false positive.

The results, as can be seen in Table 2, are satisfactory for the synthetic documents, but precision and recall get worse for real raw documents. All the measures score one for documents one to three, meaning that the GA has generated ideal regex for the cases covered by the training set. Precision and recall in document four, that has been fetched from the Web, achieves a value of one because phone numbers in this document match perfectly the pattern used by the training set ($\backslash(\backslash d+\backslash)\backslash d+\text{-}\backslash d+$).

Documents five and six have slightly worse results. Precision associated with the extraction of email records is limited by the pattern that the regex can extract,

Table 2 Extraction capacity of evolved regex. The table shows the precision (P), recall (R) and F-measure (F).

	Email	Phone	Email regex			Phone regex		
			P	**R**	**F**	**P**	**R**	**F**
Document 1	5	0	1	1	1	-	-	-
Document 2	0	5	-	-	-	1	1	1
Document 3	5	5	1	1	1	1	1	1
Document 4	0	99	-	-	-	1	1	1
Document 5	862	0	0.79	0.51	0.62	-	-	-
Document 6	88	83	0.92	1	0.96	0.8	0.8	0.8
Average			0.93	0.89	0.89	0.95	0.95	0.95

$\backslash w+@\backslash w+\backslash .com$. Using this regex, the record *name.lastname@example.com* is extracted as *lastname@example.com* and thus it generates a false positive. In addition, emails that contains special symbols and numbers that are not matched, generating also a false positive. Precision in the extraction of phone numbers in the document six has been affected by some phone numbers that contains extensions. Since the extracted record has not been correct, it has been computed as a false positive.

Recall, on the other hand, is perfect in phone numbers extraction, however it is lower in email extraction with documents five and six. In this case the limitation is imposed by emails containing a domain with more than two levels, the evolved regex in unable to extract them, thus records such as *name@it.example.com* cannot be extracted. This limitation is related to the linear nature of the coding used in the GA.

4 Conclusions and Future Work

A method to generate regex using GAs with variable-length chromosomes has been described. It has been shown that simple regex can be evolved using two sets of examples. Experiments have shown that the chromosome length automatically converges with a minimum number of parameters. Additionally, the study cases under study follows similar patterns and optimum parameters. However, there are some remarkable limitations in this approach. The most notable limitation is the linear nature of codification in GA. Coding the hierarchical regex structure with a linear chromosome yields to unnatural mapping that might be a barrier to generate complex regex, as can be seen in the evolved regex. A natural step in future research in this topic is to evolve complex regex with Grammatical Evolution.

Acknowledgements. This work has been partially supported by the Spanish Ministry of Science and Innovation under the projects Castilla-La Mancha project PEII09-0266-6640, COMPUBIODIVE (TIN2007-65989), and by HADA (TIN2007-64718).

References

Barrero, D., R-Moreno, M., López, D., García, O.: Searchy: A metasearch engine for heterogeneus sources in distributed environments. In: Proceedings of the International Conference on Dublin core and Metadata Applications, Madrid, Spain, pp. 261–265 (2005)

Barrero, D.F., Camacho, D., R-Moreno, M.D.: Automatic Web Data Extraction Based on Genetic Algorithms and Regular Expressions. In: Data Mining and Multiagent Integration, pp. 143–154. Springer, Heidelberg (2009)

Brookshear, J.G.: Theory of computation: formal languages, automata, and complexity. Benjamin-Cummings Publishing Co., Inc., Redwood City (1989)

Burke, D.S., Jong, K.A.D., Grefenstette, J.J., Ramsey, C.L., Wu, A.S.: Putting more genetics into genetic algorithms. Evolutionary Computation 6, 387–410 (1998)

Camacho, D., R-Moreno, M.D., Barrero, D.F., Akerkar, R.: Semantic wrappers for semi-structured data extraction. Computing Letters (COLE) 4(1), 1–14 (2008)

Chu, D., Rowe, J.E.: Crossover operators to control size growth in linear GP and variable length GAs. In: Wang, J. (ed.) 2008 IEEE World Congress on Computational Intelligence, Hong Kong. IEEE Computational Intelligence Society, IEEE Press, Los Alamitos (2008)

Friedl, J.E.F.: Mastering Regular Expressions. O'Reilly & Associates, Inc., Sebastopol (2002)

Gold, E.M.: Language identification in the limit. Information and Control 10(5), 447–474 (1967)

Goldberg, D.: Genetic algorithms in search, optimization and machine learning. Addison-Wesley, Reading (1989)

Holland, J.H.: Adaptation in natural and artificial systems. MIT Press, Cambridge (1992)

Hutt, B., Warwick, K.: Synapsing variable-length crossover: Meaningful crossover for variable-length genomes. IEEE Trans. Evolutionary Computation 11(1), 118–131 (2007)

Lang, K.J.: Evidence driven state merging with search (1998)

Petry, F.E., Dunay, B.D., Buckles, B.P.: Regular language induction with genetic programming. In: International Conference on Evolutionary Computation, pp. 396–400 (1994)

Ramsey, C.L., De Jong, K.A., Grefenstette, J.J., Wu, A.S., Burke, D.S.: Genome length as an evolutionary self-adaptation. In: Eiben, A.E., Bäck, T., Schoenauer, M., Schwefel, H.-P. (eds.) PPSN 1998. LNCS, vol. 1498, pp. 345–353. Springer, Heidelberg (1998)

A Case Study on Grammatical-Based Representation for Regular Expression Evolution

Antonio González-Pardo, David F. Barrero, David Camacho, and María D. R-Moreno

Abstract. Regular expressions, or simply regex, have been widely used as a powerful pattern matching and text extractor tool through decades. Although they provide a powerful and flexible notation to define and retrieve patterns from text, the syntax and the grammatical rules of these regex notations are not easy to use, and even to understand. Any regex can be represented as a Deterministic or Non-Deterministic Finite Automata; so it is possible to design a representation to automatically build a regex, and a optimization algorithm able to find the best regex in terms of complexity. This paper introduces both, a graph-based representation for regex, and a particular heuristic-based evolutionary computing algorithm based on grammatical features from this language in a particular data extraction problem.

Keywords: Regular Expressions, Grammatical-based representation, Evolutionary algorithms.

1 Introduction

Any Regular Expression, or *regex* (Friedl, 2002), can be described as a particular kind of notation for describing patterns of text. When a particular string is in the set described by a regex, it is said that the regex matches the string. The powerful pattern matching facilities provided by regex in different programming languages such as Perl, PHP, JavaScript, PCRE, Python, Ruby, or Java have not been

Antonio González-Pardo · David Camacho
Departamento de Informática. Universidad Autónoma de Madrid,
C/Francisco Tomás y Valiente, n 11, 28049 Madrid, Spain
e-mail: {antonio.gonzalez,david.camacho}@uam.es

David F. Barrero · María D. R-Moreno
Departamento de Automática. Universidad de Alcalá,
Ctra Madrid-Barcelona, Km. 33,6. 28871 Alcalá de Henares, Madrid, Spain
e-mail: {mdolores,david}@aut.uah.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 379–386.
springerlink.com

conveniently exploited by the programmers or the computer scientists due the difficulty to write and understand the syntax, as well as the semantic meaning of those regular expressions.

Any regex can be represented as a Deterministic or Non-Deterministic Finite Automata, a complete discussion of this problem can be found at (Thompson, 1968), (Kleene, 1956), (Chang and Paige, 1992). From an algorithmic and programming perspective the problem about how to represent and look for the optimal regex is an interesting, but usually a hard problem (Gold, 1978). From the set of optimization and Machine Learning methods that can be used to represent and automatically search for regex, the Evolutionary Computation (EC) (Eiben and Smith, 2008) provides a collection of algorithms inspired in the biological evolution whose characteristics make them promising to address this problem.

2 Graph-Based Representation for Regex

Different approaches can be defined to represent a regex as an evolutionary based individual, from GAs (Barrero et al., 2009) to GP (Dunay et al., 1994). The final representation of this individual is a critical aspect in any EC algorithm. This work presents an initial approach based on graphs that uses the basic and syntactic considerations of the regex notation.

2.1 A Brief Introduction to Regular Expressions

Regular expressions (Cox and Russ, 2007) are a tool used for providing a compact representation of string patterns. They have been widely used by programmers and systems administrators since most of common languages such as Perl, Java, C or PHP support regex and many UNIX administration tools such as *grep* or *sed* use them. Regex are commonly used, for instance, to extract strings or validate user inputs. Regex contain some special symbols called wildcards. Following the basic wildcards from IEEE POSIX Basic Regular Expressions (BRE) standard, and POSIX Extended Regular Expressions (ERE) notation, 4 wildcards[1] are considered in this work which are *Plus*, *Star*, *Question mark* and *Pipe*. These wildcards are explained bellow and a graph-based representation are provided in Figure 1:

- **Plus +.** Repeats the previous item once or more.
- **Start *.** Repeats the previous item zero or more times.
- **Question mark ?:** Makes the preceding item optional.
- **Pipe |:** Causes the regex engine to match either the part on the left side, or the part on the right side. Can be strung together into a series of options.

[1] A complete syntax reference can be found at `http://www.regular-expressions.info/reference.html`

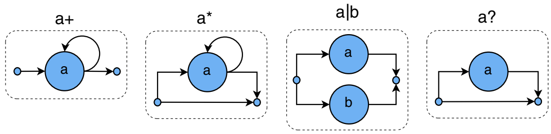


Fig. 1 Graph-based representation for main wildcards

2.2 A Particular Example on Web Data Extraction

The extraction of simple web data, as URL addresses, is a very common and useful task achieved by a huge type of programs that crawl and search the World Wide Web for useful information. Using the representation shown in previous section, any regex could be represented as a Finite Automata. Next regex could be used to retrieve a generic URL link (it only considers most current usual protocols as http, https or ftp) from a web page:

(https|http|ftp)://[a-zA-Z]+(\.[a-zA-Z]+)*\.(com|net|org)

Figure 2 shows in detail the graphical representation of this regex. Note that the aim of the graph is to accept the string that matches the regex, not to accept the string that represents the regex. The graph must accept, for example, "*http://my.url.com*" but not the regex itself. Also note the initial node is represented as a node with a double border and without any label.

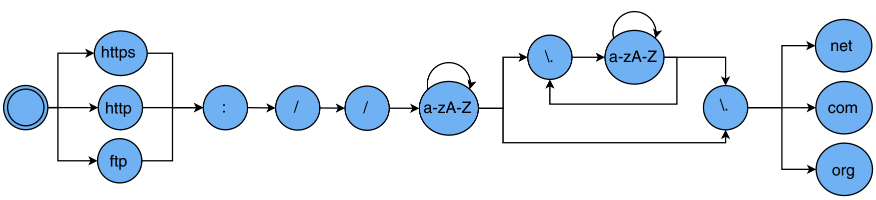


Fig. 2 Graph-based representation for a regex

2.3 Nodes Representation

A major problem in a graph representation is the combinatorial explosion that might happen if each character is represented by a node. In order to avoid it, the representation must reduce the size of the graphs using nodes to represent composed symbols instead of simple ones. In this way the search space is drastically reduced, however a new problem arises: how to automatically construct the composed alphabet.

Our approach uses the Zipf Law (Zipf, 1935) to construct the alphabet. Zipf observed that given a corpus, a small subset of the words concentrates a high frequency of occurrences, while most words appears few times in the corpus. This fact is used

Table 1 Categorization of Nodes

Node	Elements
N_1	{:, /, \.}
N_2	{http, https, ftp, com, es, org, net, 8080}
N_3	{\d, \w, a-z, A-Z, |}
N_4	{(,), {, }, [,]}
N_5	{*, + , ?}

to design a heuristic able to identify important strings, and use it to build the alphabet. For instance, given a set of URL examples, it is possible to identify ”http” and ”ftp” as common strings, and thus use them as a symbol.

The graph that is evolved is composed by three types of nodes. Each node is able to accept a composed symbol such as ”http”. Nodes of type N_1 contain the symbols that generated Zipf valid strings, while nodes of type N_2 are the strings extracted using Zipf Law. Symbols N_3, N_4 and N_5 are the especial characters from regex with an increasing semantics, so the wildcard ”*” represents a cyclic graph (see previous section), whereas symbol ”(” is used to maintain the syntax correctness of the regex. The information given from these kind of nodes will be used by an evolutionary algorithm to guide both the mutation and crossover operations.

2.4 Population Representation and Initial Population Generation

Each individual in the population represents a graph whose phenotype is a regex. The graph is represented by a variable length list of interconnected nodes. In order to generate always a syntactically correct regex the set of N_5 nodes has been extended with three more elements, which are *+’*, **’* and *?’*. These nodes do not belong to the vocabulary about regex nor the vocabulary about URL, but they are needed to restrict the effect of the elements +, * and *?*, respectively. For example, the regex *abc(de)** will be represented as *abc*(de)*’*, and the program will know that only the string *ed* is affected by the operator *.

Figure 3 shows the genotype of an individual that represents the regex *(http|ftp):(\w)**.

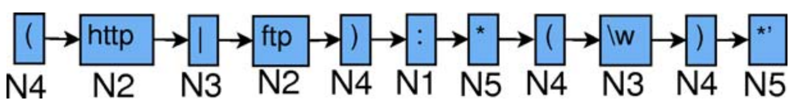


Fig. 3 Regular expression for URL address represented as a list of N_j nodes

Once an individual can be represented, it has been defined some heuristics to guide the generation of the initial population. These heuristics allow to create well-formed (syntactically and grammatically) individuals. In order to do that, a maximum length of the genotype has been defined. Note, that this length is a constraint for the generation of the initial population, for that reason its possible to obtain, in any generation, any individual whose genotype exceeds this maximum length. A manually generated regex able to accept all the patterns of this study has 55 nodes. The maximum length of the genotype of the initial population will be 70 nodes. This value allows initial population to exceed the length of the valid regex in, approximately, a 25%.

3 Mutation and Crossover Operations for Regex Evolution

From our previous nodes classification, two different types of mutation have been defined:

- **Random.** This is equivalent to the uniform mutation operator found in GA or point mutation in GP. Once a particular node i that belongs to class j ($n_{i,j}$) in the graph is selected, it is changed by any randomly token available in our sets of nodes. Therefore, any node in the graph can be substituted by any other.
- ***Guided.*** This mutation takes randomly the node $n_{i,j} \in N_j$ from the graph, and it only can be changed by other token (randomly selected) which belongs to the class N_j. This guided mutation tries to use the statistical knowledge acquired from the analysis of natural language, so the token **http** could be mutated into **ftp**, and the grammatical knowledge from the regex notation, so the wildcard ***** could be mutated into **+**. Note, that in the case of the node belongs to class N_4 and N_5, the same mutation has to be performed twice due to nodes belonging to these classes appear in pairs. That means, if a node * is mutated into **+**, then the corresponding node ***’**, which represent the end of the operator, must change into **+’**.

The guided mutation allows one to reuse grammatical and syntactical knowledge from the language used to generate regex, it allows to easily generate correct individuals. However, this produces a second important bias in the searching process of individuals.

The crossover operation using a graph and grammatical-based representation (see section 2) needs to consider the inner structure of this graph to select part of the graphs that can be later used in the offspring generation. To compare how the representation works, some crossover operations have been defined:

- **One-point.** This is the equivalent crossover of the traditional one-point crossover found in GA. A point is randomly selected from both parents splitting its genotype in two parts. The new individual generated will inherit two parts, one from each parent.
- **Two-point.** Two points inside the graph (both randomly) are selected, then both sections of the graph are interchanged. It is the equivalent of two-points GA crossover.
- **Grammatical-based selection.** In order to explain this operation, a new concept *typed block* is defined. Let be B_x, where $x \in \{N_4, N_5\}$, a block of variable length limited by nodes of class x, B_x is named as *block of type x*. For example:

+(http|ftp)+' is a block limited by nodes belonging to N_5 class, thus is a B_5 block. Grammatical-based selection is heuristically guided and it works as follows:

1. One point in the graph is selected.
2. If the selected node is contained into a B_4 block, then the entire block is selected.
3. If the selected node is not included into a B_4 block, the algorithm will try to determine the B_5 block that contains the selected node. If that block exists, then it is selected.
4. Finally, if the selected node does not belong to a B_5 block nor a B_4 block, then only the selected node will be exchanged.

4 Experimental Results

Genetic Algorithms (GA) need an objective function which allows to evaluate the population and to differentiate its individual. This function is called *fitness function*. In order to evaluate each regex generated by the program, a file with different URLs is used. With this two notions and the context of this case study, it is easy to identify that a possible fitness function could be the number of URLs that the generated regex is able to match. Nevertheless, this fitness function is very restrictive and this makes the evolution of the individual difficult.

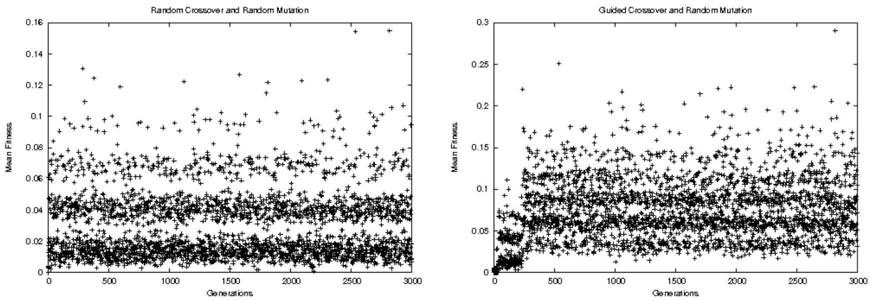
The fitness function used in this case study is based on the proportion of character that the regex is able to match from the whole test file. Given an individuals i, the fitness of i is defined by equation 1.

$$\mathfrak{F}(i) = \frac{\sum_{j=1}^{N} \frac{m_{ij}}{l_j}}{N} \tag{1}$$

Where N is the number of URLs in the test file, m_{ij} is the number of characters that the individual i matches from the URL number j and l_j is the total length of the URL number j.

There are some parameters which must be specified to execute the GA, these parameters are the number of parents in the population, the number of offspring population and the mutation rate. These values have been set with the following values: the parent population is 30, the offspring population is 45 and the mutation rate is $0,05$. With these values, the program has been executed taking into account all possible combinations of Mutation and Crossover operations, that means trying to launch the program using random crossover and random mutation, guided crossover and random mutation, random crossover and guided mutation and, guided crossover and guided mutation.

Figure 4(a) shows the results of launching the program with random crossover and random mutation. As it can be seen, there is no evolution in the population due to data dispersion. These bad results are expected because with random mutation and random crossover, there is no guarantee of generating well-formed individual. Therefore, the probability of generating any individual with a good fitness value is very low. Another execution uses guided crossover and random mutation in order to



(a) Random crossover and random mutation

(b) Guided crossover and random mutation

Fig. 4 Mean fitness of graph evolution

obtain better results than the previous one. The results of this execution are shown in Figure 4(b). As it can be seen, this experiment provides better results than the previous one, but they are not useful since it only recognizes the 30% of the examples. The experiment with random crossover and guided mutation does not work due to guided mutation needs the creation of well-formed individual (syntactically and grammatically), and this restriction is not always true by the random crossover.

Finally, with guided mutation and guided crossover the best result are obtained. All the generated individuals are well-formed (syntactically and grammatically) due to the heuristic applied in the operators (see Section 3). Figure 5 shows the result of this experiment. The evolution of the population is represented in the increase of the fitness of the population for each generation. The chart shows how the mean fitness

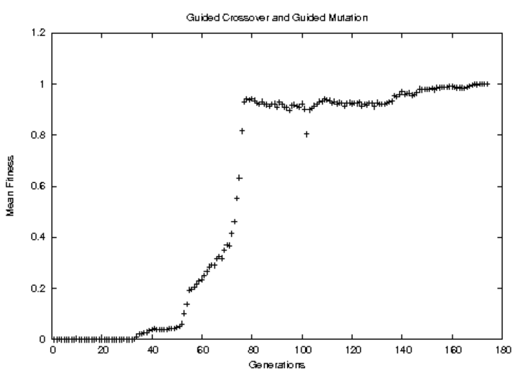


Fig. 5 Results of execution with guided crossover and guided mutation

increases and it is closer to the optimal value. This value is 1 which represents that it is matched a 100% of the examples in the test file. An example of individual generated by the program is the following: $[\backslash w * | ftp * \backslash d/?/A-Z]$.

5 Conclusions

This work shows a representation and evolutionary algorithm case study based on both, graphs and grammatical features from the syntax and grammar standards used to define regex. It have been defined two different kind of specific evolutionary operators, mutation and crossover guided by these grammatical considerations. Although the experimental results provide an initial evaluation of the genetic algorithm, more experiments must be carried out in the future and other parameters, such as precision, need to be considered in order to measure the efficiency of the algorithm.

Acknowledgements. This work has been partially supported by the Spanish Ministry of Science and Innovation under the projects Castilla-La Mancha project PEII09-0266-6640, COMPUBIODIVE (TIN2007-65989), and by HADA (TIN2007-64718).

References

Barrero, D.F., Camacho, D., R-Moreno, M.D.: Automatic Web Data Extraction Based on Genetic Algorithms and Regular Expressions. In: Data Mining and Multiagent Integration. Springer, Heidelberg (2009)

Chang, C.-H., Paige, R.: From regular expressions to dfa's using compressed nfa's, pp. 90–110 (1992)

Cox, R. (ed.): Regular expression matching can be simple and fast (2007)

Dunay, B.D., Petry, F., Buckles, B.P.: Regular language induction with genetic programming. In: Proceedings of the 1994 IEEE World Congress on Computational Intelligence, Orlando, Florida, USA, pp. 396–400. IEEE Press, Los Alamitos (1994)

Eiben, A.E., Smith, J.E.: Introduction to Evolutionary Computing. Natural Computing Series. Springer, Heidelberg (2008)

Friedl, J.E.F.: Mastering Regular Expressions. O'Reilly & Associates, Inc., Sebastopol (2002)

Gold, E.M.: Complexity of automaton identification from given data. Inform. Control 37, 302–320 (1978)

Kleene, S.C.: Representation of events in nerve nets and finite automata. In: Shannon, C.E., McCarthy, J. (eds.) Automata studies, vol. 34, pp. 3–40 (1956)

Thompson, K.: Regular expression search algorithm. Comm. Assoc. Comp. Mach. 11(6), 419–422 (1968)

Zipf, G.: The psycho-biology of language. Houghton Mifflin, Boston (1935)

Towards the Automatic Programming of H Systems: jHsys, a Java H System Simulator

J.M. Rojas Siles, M. de la Cruz Echeandía, and A. Ortega de la Puente

Abstract. The main goal of this paper is to describe how we consider that splicing systems (a family of abstract bio-inspired computing devices) can be automatically programmed (designed) in the future. One of the necessary steps is to formally describe the computer being programmed (splicing systems). Some of the authors of this paper have previously solved this problem. Another necessary step is to develop a simulator for H systems. We propose applying Christiansen Grammar Evolution (an evolutionary automatic programming algorithm developed by the authors) to complete the process. This technique includes a fitness function that the simulator requires. This paper is devoted to describe jHSys, a Java simulator for splicing (H) systems.

1 Motivation

The task of automatically writing programs can be seen as a search problem: finding the best in a set of candidate programs automatically generated. Any general search technique can be used to solve this problem.

Conventional personal computers are based on the well known von Neumann architecture, that tries to implement the Turing machine by means of electronic devices: aritmetic and logic unit, registers, data and direction buses, etc. A great effort is being dedicated to one of the current topics of interest of our research group: the design of new abstract computing devices which can be considered as alternative architectures to design new families of computers. Some of them, inspired in the way used by Nature to efficiently solve difficult tasks, are called *natural or unconventional computers*. A few of the natural phenomena inspiring these devices are:

J.M. Rojas Siles
Departamento de Lenguajes y Sistemas Informáticos e Ingeniería de Software, Universidad Politécnica de Madrid
e-mail: `josemiguel.rojas@upm.es`

M. de laCruz Echeandía · A.O. de la Puente
Departamento Ingeniería Informática, Universidad Autónoma de Madrid
e-mail: `marina.cruz@uam.es, alfonso.ortega@uam.es`

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 387–394.
springerlink.com

the role of membranes in behaviour of cells, the structure of genetic information, and the way in which species evolve.

Any computer scientist has a clear idea about how to program *conventional* (von Neumann) computers by means of different high level programming languages and their corresponding compilers, which translate programs into machine code. On the other hand, imagining how to program unconventional computers is quite difficult.

One of the main topics of interest of our research group is the formal specification of complex systems that makes it possible to apply formal tools to their design, or study some of their properties. We have successfully applied this approach to different bio-inspired computational devices (L systems, cellular automata [2, 3, 4, 5]) and proposed a new evolutionary automatic programming algorithm (Christiansen Grammar Evolution or CGE [6]) as a powerful tool to design complex systems to solve specific tasks.

CGE wholly describes the candidate solutions, both syntactically and semantically, by means of Christiansen grammars. CGE improves the performance of other approaches, because it reduces the search space by excluding non-promising individuals with syntactic or semantic errors.

Splicing systems are abstract devices with a complex structure, because some of their components depends on others. This dependence makes it difficult to use genetic techniques to search splicing systems because, in this circumstance, genetic operators usually produce a great number of incorrect individuals (both syntactically and semantically).

This paper is focused on one of the steps needed for using CGE to automatically program splicing systems: their simulation.

Figure 1 graphically describes the different blocks which can be considered to propose a general way to program natural computers similar to splicing systems to solve a given problem.

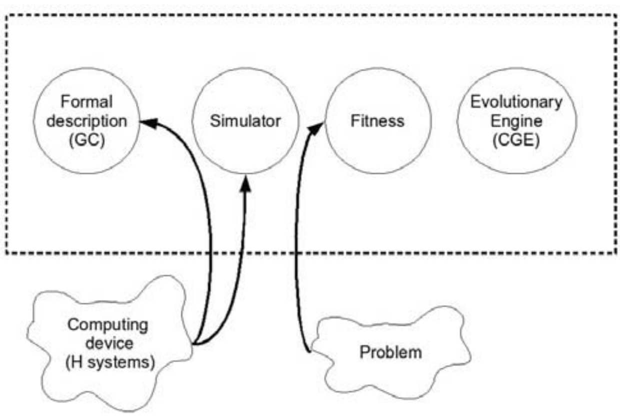


Fig. 1 Blocks of a general way to program natural computers

This method takes as inputs the following elements: (1) the *target problem* to be solved and (2) the *computing device* that will be used to solve the problem.

The method consists of the following modules: (1) An *evolutionary engine*, used as an automatic programming algorithm. This engine has to handle candidate solutions with a complex structure. We propose using Christiansen Grammar Evolution. (2) A *formal description of the computing device* being programmed. In [1] the authors design a Christiansen grammar for universal splicing systems. (3) A *simulator for the computing device* that will be used to compute the fitness function. The current paper is focused on this module. (4) The *fitness function*, which must fulfill two roles: simulating the generated solution (in this case, a particular splicing system) and measuring how well the solution solves the target problem.

In the following sections, we will briefly introduce splicing systems, and describe jHsys, a Java simulator for splicing (H) systems. Finally, conclusions and further research lines are discussed.

2 Introduction

2.1 Introduction to Splicing Systems

Splicing systems were introduced by Head in [7] as a DNA inspired computing device. Splicing systems formalize the DNA recombination operation.

A splicing rule on strings is formally defined by means of four patterns u_1, u_2, u_3 and u_4 and, when applied to two strings $x = x_1u_1u_2x_2$ and $y = y_1u_3u_4y_2$, it can produce two resulting strings $z = x_1u_1u_4y_2$ and $z' = y_1u_3u_2x_2$; although the second one (z') usually is discarded because the simmetric rule has the same result.

Formally they are represented as $u_1\#u_2\$u_3\#u_4$.

A splicing system (H) consists of a set of splicing rules (R) that is applied to a set of strings (L), both sets share the same alphabet (V).

Extended splicing systems are one of the best studied variants of the basic model. They use a distinguished subset of the alphabet (T) named *set of terminals* with the same meaning than in Chomsky grammars.

Formally $H = (V,T,L,R)$.

The *language generated by* a splicing system ($L(H)$) contains all the possible results of the splicing rules to the initial language (L) The languages generated by extended splicing systems contain only strings of terminal symbols.

For example. It is easy to demonstrate that the following system generates strings with the structure $c_1(ab)^n(ab)^m...c_2$

$$H_0 = \{V_0, L_0, R_0\}$$

- $V_0 = \{a, b, c_1, c_2, c_3, c_4\}$
- $L_0 = \{c_1abc_2, c_3ac_4, c_4bc_3, c_1c_4, c_4c_2\}$
- $R_0 = \{c_3a\#c_4\$c_1\#a^+b^+c_2, c_3a^+b^+\#c_2\$c_4\#bc_3, c_1\#c_4\$c_3\#a^+b^+c_3,$ $c_1a^+b^+\#c_3\$c_4\#c_2, c_1a^+b^+\#\$c_1\#a^+b^+c_2\}$

The complexity of a splicing system can be studied in terms of the complexity of the initial set and the complexity of the rules in the Chomsky Hierarchy. It is known [8] that the extended splicing system with a finite initial language and with rules whose patterns are specified using regular expressions is equivalent to Turing machines.

2.2 jHSys: A Java Splicing Systems Simulator

2.2.1 jHSys Design

jHSys stands for java H System Simulator. It is a multi-threaded Java application to simulate an ample set of families of splicing systems, including those introduced in [8] (regular extended H systems). jHSys allows the specification of a splicing system H in terms of its main components (an alphabet V, an initial language L and a set of splicing rules R).

jHSys fulfills all the features of the original splicing systems model. Moreover, the design of our simulator allows us to easily scale it to support new variants of this kind of systems. An example of this characteristic is the possibility of defining a set of terminals ($T \subseteq V$) to simulate the aforementioned family of extended splicing systems. The simulator can also deal with different stopping conditions, to determine when a particular simulation should stop. Thus, a set of stopping conditions S is added to the initial model. The way of specifying these conditions is shown later on this paper.

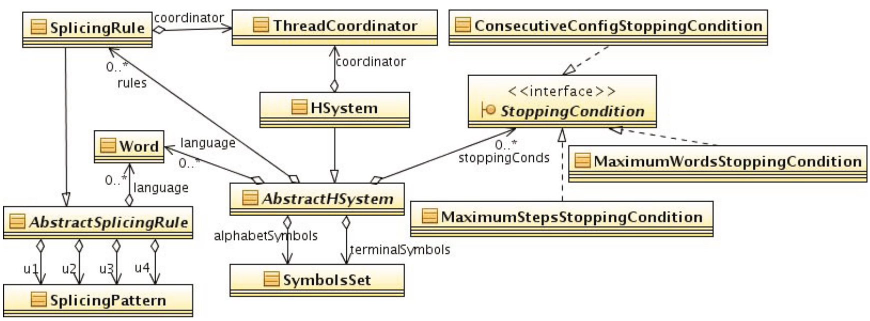


Fig. 2 jHSys Class Diagram

Figure 2 shows a simplified class diagram of jHSys. For the sake of simplicity, the attributes and methods implemented by the different classes are omitted.

We have used a Java interface to represent stopping conditions. This technique will ease the future inclusion of new kinds of conditions. The class `SplincingRules` implements the Java interface `Runnable`. A different Java thread is in this way assigned to each rule to improve the efficiency of the simulator on parallel platforms. The `ThreadCoordinator` class properly synchronizes the threads.

jHSys simulations can be summarized as follows:

0. The user has to describe the system under consideration in an XML file.
1. This input file is translated into the corresponding Java data structures.
2. Until computation ends:

   1 An `APPLY_RULE` *step* is taken to verify and execute all the applicable splicing rules.
   2 An `UPDATE_LANGUAGE` *step* collects all the new words added in the previous step and includes them in the resulting language of the system.
   3 The stopping conditions are evaluated.

2.2.2 XML Specification Files

As we have previously mentioned, the input of jHSys is an XML file describing the system to be simulated. Figure 3 shows the XML code corresponding to the splicing system example previously described:

```
<?xml version="1.0"?>
<!-- H-System specification file -->
<HSYSTEM>
   <ALPHABET symbols="a b c1 c2 c3 c4"/>
   <TERMINALS symbols="a b c1 c2 c3 c4"/>
   <LANGUAGE words="c1abc2 c3ac4 c4bc3 c1c4 c4c2"/>
   <SPLICING_RULES>
      <SPLICING_RULE u1="c3a" u2="c4" u3="c1" u4="a+b+c2"/>
      <SPLICING_RULE u1="c3a+b+" u2="c2" u3="c4" u4="bc3"/>
      <SPLICING_RULE u1="c1" u2="c4" u3="c3" u4="a+b+c3"/>
      <SPLICING_RULE u1="c1a+b+" u2="c3" u3="c4" u4="c2"/>
      <SPLICING_RULE u1="c1a+b+" u2="" u3="c1" u4="a+b+c2"/>
   </SPLICING_RULES>
   <STOPPING_CONDITION>
      <CONDITION type="MaximumWordsStoppingCondition" maximum="12"/>
   </STOPPING_CONDITION>
</HSYSTEM>
```

Fig. 3 XML file for the example

- The main tag of the xml structure is `<HSYSTEM>`, that contains the whole description of the splicing system under consideration.
- The tag `<ALPHABET>` declares the available symbols.
- The tag `<TERMINALS>` contains, when simulating an extended splicing system, the set of terminal symbols. As a prerequisite of the simulation, jHSys will check whether it is a subset of the alphabet or not.
- The tag `<LANGUAGE>` contains the list of words of the initial language of the system. All their symbols must belong to the set defined by `<ALPHABET>`.
- The tag `<SPLICING_RULES>` contains all the splicing rules of the splicing system. Each rule is composed of four patterns ($u1$, $u2$, $u3$ and $u4$), which the user must define as Java regular expressions. For example, the splicing rule formally defined as $(c_3a\#c_4\$c_1\#a^+b^+c_2)$ is translated into the XML component `<SPLICING_RULE u1="c3a" u2="c4" u3="c1" u4="a+b+c2"/>`.

- Stopping conditions are specified within the tag `<STOPPING_CONDITIONS>`. jHSys supports three kinds of stopping conditions:
  - **MaximumWordsStoppingCondition:** It stops the simulation when the resulting language contains a certain number of words. The following fragment shows the XML tag for 12 words
    ```
    <CONDITION type="MaximumWordsStoppingCondition"maximum="12"/>
    ```
  - **MaximumStepsStoppingCondition:** Simulation stops after a given number of steps. The following tag corresponds to 15 steps:
    ```
    <CONDITION type="MaximumStepsStoppingCondition"maximum="15"/>
    ```
  - **ConsecutiveConfigStoppingCondition:** Simulation finishes when there is no change in the resulting languages between two consecutive `APPLY_RULE` steps. Its representation in the xml file is
    ```
    <CONDITION type="ConsecutiveConfigStoppingCondition"/>
    ```

2.2.3 Running jHSys

There are two versions of jHsys. Either can be run by invoking it from the command line or by using an Eclipse plug-in. Eclipse [9] is a free integrated developing environment. It provides the user with a set of tools that includes a rich XML editor. To use jHSys with Eclipse, the user must download, install the plug-in (available online at *http://www.clip.dia.fi.upm.es/∼jmrojas/*), and use the provided toolbar options. In this case, the user can benefit from the features of the built-in XML editor to create and modify the files that describe the splicing systems being simulated. The *console view* of Eclipse will contain the output of the simulations.

The standalone version of the simulator is also available at *http://www.clip.dia.fi.upm.es/∼ jmrojas/*, and it is invoked by the command

```
jhsys [inputFile.xml] [outputFile.log]
```

or

```
java -jar jhsys [inputFile.xml] [outputFile.log]
```

Where

- `inputFile.xml` describes the splicing system under consideration.
- `outputFile.log` is a log file that will store the output of the simulation:
  - The first lines briefly describe the splicing system.
  - There is a block for each `APPLY_RULE` step with a line for each applied rule with the list of words it adds to the resulting language.
  - There is a block for each `UPDATE_LANGUAGE` step with the current language.

  Figure 4 shows the log file corresponding to the splicing system we have previously described.

After two iterations (a total of four steps) the `MaximumWordsStoppingCondition` is reached, i.e., the resulting language contains 12 words. At this moment the simulation stops and the result is the language generated by the second

```
XML FILE LOADED AND PARSED SUCCESSFULLY...
H-SYSTEM INFO PARSED SUCCESSFULLY...
STOPPING CONDITIONS INFO PARSED SUCCESSFULLY...
H-SYSTEM XML PARSED SUCCESSFULLY...
RUNNING H-SYSTEM...
*** H-SYSTEM INITIAL CONFIGURATION ***
ALPHABET: [a, b, c1, c2, c3, c4]
TERMINALS: [a, b, c1, c2, c3, c4]
LANGUAGE: [c1abc2, c3ac4, c4bc3, c1c4, c4c2]
SPLICING RULES:
Splicing Rule 0 : [u1="c3a", u2="c4", u3="c1", u4="a+b+c2"]
Splicing Rule 1 : [u1="c3a+b+", u2="c2", u3="c4", u4="bc3"]
Splicing Rule 2 : [u1="c1", u2="c4", u3="c3", u4="a+b+c3"]
Splicing Rule 3 : [u1="c1a+b+", u2="c3", u3="c4", u4="c2"]
Splicing Rule 4 : [u1="c1a+b+", u2="", u3="c1", u4="a+b+c2"]
STOPPING CONDITIONS:
MaximumWords=12
*** APPLY_RULE - STEP 1 ***
Rule 2 adds new words: []
Rule 0 adds new words: [c3aabc2, c1c4]
Rule 1 adds new words: [c3aabbc3, c4c2]
Rule 3 adds new words: []
Rule 4 adds new words: [c1ababc2, c1c2]
*** UPDATE_LANGUAGE - STEP 2 ***
LANGUAGE: [c1abc2, c3ac4, c4bc3, c1c4, c4c2, c3aabc2, c3aabbc3,
c1ababc2, c1c2]
*** APPLY_RULE - STEP 3 ***
Rule 1 adds new words: [c3aabbc3, c4c2]
Rule 2 adds new words: [c1aabbc3, c3c4]
Rule 4 adds new words: [c1ababc2, c1c2, c1ababc2, c1abc2, c1aababc2,
c1bc3, c1aabbabc2, c1c3]
Rule 0 adds new words: [c3aabc2, c1c4]
Rule 3 adds new words: [c1aabbc2, c4c3]
*** UPDATE_LANGUAGE - STEP 4 ***
LANGUAGE: [c1abc2, c3ac4, c4bc3, c1c4, c4c2, c3aabc2, c3aabbc3,
c1ababc2, c1c2, c1aabbc3, c3c4, c1aababc2, c1bc3, c1aabbabc2,
c1c3, c1aabbc2, c4c3]

Stopping condition found: HSystems.stopping.MaximumWordsStoppingCondition
H-System has stopped!!!
```

Fig. 4 Contents of the log file for the studied splicing system

`UPDATE_LANGUAGE` step (the fourth step). Figure 4 shows the complete output for this example.

3 Conclusions and Future Research

This paper describes a Java simulator for an ample set of splicing systems that includes some well-known universal H systems. The simulator reads the description of the systems from an XML file and stores the results in a log file. Some of the authors of this paper have previously designed a Christiansen grammar for the same families of splicing systems. In the future, we plan to study the possibility of proposing a methodology to automatically design splicing systems to solve given problems by means of Christiansen Grammar Evolution (an evolutionary automatic programming algorithm proposed by the authors of this paper).

In order to reach this goal, we will need to follow the following steps:

- Selecting a particular problem to be solved with splicing systems.
- Particularizing the Christiansen grammar that we have previously designed according to the particular family that can solve this problem and developing a Java version of the grammar that can be handled by the proposed algorithm.
- Developing a fitness function with the simulator described in this paper.
- Designing a set of experiments to select the proper splicing system.

Acknowledgements. This work was partially supported by DGUI CAM/UAM, project CCG08-UAM/TIC-4425. J. M. Rojas Siles is supported by the Spanish Ministry of Science and Innovation under the grant BES-2009-027467, associated to the TIN2008-05624 "DOVES" project. The authors thank Dr. Manuel Alfonseca for his help while preparing this document.

References

1. de la Cruz, M., Ortega, A.: A Christiansen grammar for universal splicing systems. In: Mira, J., Ferrández, J.M., Álvarez, J.R., de la Paz, F., Toledo, F.J. (eds.) IWINAC 2009. LNCS, vol. 5601, pp. 336–345. Springer, Heidelberg (2009)
2. Ortega, A., Dalhoum, A.A., Alfonseca, M.: Grammatical evolution to design fractal curves with a given dimension. IBM Jr. of Res. and Dev. 47(4), 483–493 (2003)
3. Ortega, A., Abu, D.A.L., Alfonseca, M.: Cellular Automata equivalent to DIL Systems. In: Proceedings of 5th Middle East Symposium on Simulation and Modelling (MESM 2003), Eurosim, 5-7 Ene. 2004, pp. 120–124. Emiratos Arabes Unidos. Pub., Sharjah (2003) ISBN: 90-77381-06-6
4. Abu, D.A.L., Ortega, A., Alfonseca, M.: Cellular Automata equivalent to PD0L Systems. In: Proceedings of International Arab Conference on Information Technology (ACIT 2003), December 20-23, pp. 819–825. Egipto. Pub., Alexandria (2003)
5. Abu, D.A.L., Ortega, A., Alfonseca, M.: Cellular Automata equivalent to D0L Systems. In: 3rd WSEAS International Conference on Systems Theory and Scientific Computation, Special Session on Cellular Automata and Applications, Proceedings en CDROM, November 15-17. Grecia. Pub., Rodas (2003)
6. Ortega, A., de la Cruz, M., Alfonseca, M.: Christiansen Grammar Evolution: grammatical evolution with semantics. IEEE Transactions on Evolutionary Computation 11, 77–90 (2007)
7. Head, T.: Splicing schemes and DNA. In: Rozenberg, G., Salomaa, A. (eds.) Lindenmayer Systems; Impacts on Theoretical Computer Science, Computer Graphics, and Developmental Biology, pp. 371–383. Springer, Berlin (1992)
8. Paun, G.: Regular extended H systems are computationally universal. Journal of Automata, Languages, Combinatorics 1(1), 27–36 (1996)
9. `http://www.eclipse.org/`

L Systems as Bio-MAS for Natural Language Processing

Leonor Becerra-Bonache, Suna Bensch, and M. Dolores Jiménez-López

Abstract. In this paper, we claim that Lindenmayer systems (L systems) –more precisely, ET0L systems– can be considered as *bio-inspired multi-agent systems* that, because of its inherent features, can be usefully applied to the field of natural language processing (NLP). L systems are a biologically inspired branch of the field of formal languages that provide a *parallel* and non-sequential grammatical formalism and that can be expressed as a multi-agent system. Taking into account these features and the benefits of the multi-agent approach to NLP, we propose to apply L systems to the description, analysis and processing of natural languages.

1 Introduction

In general, multi-agent systems offer strong models for representing complex and dynamic real-world environment. 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 and useful set of structures and processes for designing and building complex software applications. 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. The metaphor

Leonor Becerra-Bonache
Research Group on Mathematical Linguistics, Universitat Rovira i Virgili, 43002 Tarragona, Spain
e-mail: leonor.becerra@urv.cat

Suna Bensch
Institutionen för Datenvetenskap, Umeå Universitet, 90187 Umeå, Sweden
e-mail: suna@cs.umu.se

M. Dolores Jiménez-López
Research Group on Mathematical Linguistics, Universitat Rovira i Virgili, 43002 Tarragona, Spain
e-mail: mariadolores.jimenez@urv.cat

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 395–402.
springerlink.com

of autonomous problem solving entities cooperating and coordinating to achieve their objectives is a natural way of conceptualizing many problems. The multi-agent system literature spans a wide range of fields including robotics, mathematics, linguistics, psychology, and sociology, as well as computer science. Multi-agent systems promote the interaction and cooperation of autonomous agents in order to deal with complex tasks. This architecture has the advantage of distributing a hard task among several task-specific agents that collaborate in the solution of the problem. In a multi-agent system, agents cooperate to achieve a common objective. This idea seems to be very adequate in the field of natural language processing. In fact, the task of processing natural language requires multiple agents working together in the pursuit of a common goal: the *generation/understanding of language*. Some authors have suggest the possibility of using multi-agent systems in natural language processing to represent cooperation among distinct linguistic levels [10, 6, 17, 16, 12, 2]. With the idea of defending the adequacy of multi-agent systems in the field of natural language processing, we propose the application of Lindenmayer Systems to the description, analysis and processing of natural language. L systems can be seen as a biologically inspired branch of the field of formal languages that provide a *parallel* and non sequential grammatical formalism. In this paper, we claim that L systems –more precisely, ET0L systems– can be considered as *bio-inspired multi-agent systems* that, because of its inherent features, can be usefully applied to the field of natural language processing.

The paper is organized as follows. Section 2 introduces Lindenmayer systems and the basic definition of ET0L systems. Section 3 expresses L systems as a bio-inspired multi-agent systems. Section 4 presents the application of L systems to natural language processing. Final conclusions are presented in section 5. Throughout the paper, we assume the reader to be familiar with basic notions in the theory of formal languages. For more information the reader is referred to [15].

2 Lindenmayer Systems

Aristid Lindenmayer introduced, in 1968 [13], specific rewriting systems as models of developmental biology, which today are called Lindenmayer systems or L systems. L systems model biological growth and because growth happens in multiple areas of an organism, growth is parallel. The essential difference between Chomsky grammars and L systems lies in the method of applying productions. In Chomsky grammars productions are applied sequentially, whereas in L systems they are applied in parallel and simultaneously replace all letters in a given word. This difference reflects the biological motivation of L systems. Productions are intended to capture cell divisions in multicellular organisms, where many divisions may occur at the same time. The investigations of L systems are an important and wide area in the theory of formal languages. The modelling of different environmental influences, for example, growth during day versus night, lead to different L systems and thus to different L languages. The study of L languages has resulted in a language hierarchy, namely the L system hierarchy. Lindenmayer systems are well investigated parallel rewriting systems. For an overview see [11] and for the mathematical theory of L systems see [14].

Definition 1. *An extended tabled Lindenmayer system without interaction (ET0L system, for short) is a quadruple $G = (\Sigma, H, \omega, \Delta)$, where Σ is the alphabet, Δ is the terminal alphabet, $\Delta \subseteq \Sigma$, H is a finite set of finite substitutions from Σ into Σ^*, and $\omega \in \Sigma^*$ is the axiom.*

Definition 2. *For x and y in Σ^* and $h \in H$, we write $x \underset{h}{\Longrightarrow} y$ if and only if $y \in h(x)$. A substitution h in H is called a* table.

Definition 3. *The language generated by G is defined as:*
$L(G) = \{w \in \Delta^* \mid \omega \underset{h_{i_1}}{\Longrightarrow} w_1 \underset{h_{i_2}}{\Longrightarrow} \dots \underset{h_{i_m}}{\Longrightarrow} w_m = w$ *for some* $m \geq 0$ *and* $h_{i_j} \in H$ *with* $1 \leq j \leq m\}$.

By $\mathscr{L}$(ET0L) we denote the family of ET0L languages.

Example 1. Let $G_1 = (\{A,B,C,a,b,c\}, \{h_1,h_2\}, ABC, \{a,b,c\})$ be an ET0L system, where h_1 and h_2 are given as follows:

$$h_1 = \{A \rightarrow aA, B \rightarrow bB, C \rightarrow cC, a \rightarrow a, b \rightarrow b, c \rightarrow c\},$$
$$h_2 = \{A \rightarrow a, B \rightarrow b, C \rightarrow c, a \rightarrow a, b \rightarrow b, c \rightarrow c\}.$$

The axiom ABC can be rewritten using the first table h_1 or the second table h_2. Using the first three productions $A \rightarrow aA$, $B \rightarrow bB$, $C \rightarrow cC$ in h_1 adds a symbol a, b, and c, respectively in every derivation step. Using table h_2 terminates the derivation process. Consider the derivation of the word $a^2b^2c^2$:

$$ABC \underset{h_1}{\Longrightarrow} aAbBcC \underset{h_2}{\Longrightarrow} aabbcc$$

The language generated is $L(G_1) = \{a^nb^nc^n \mid n,m \geq 1\}$.

Example 2. Let $G_2 = (\{A,B,a,b\}, \{h_1,h_2,h_3,h_4\}, AA, \{a,b\})$ be an ET0L system, where the tables are given as follows:

$$h_1 = \{A \rightarrow aA, B \rightarrow B, a \rightarrow a, b \rightarrow b\},$$
$$h_2 = \{B \rightarrow A, A \rightarrow B, a \rightarrow a, b \rightarrow b\},$$
$$h_3 = \{B \rightarrow bB, A \rightarrow A, a \rightarrow a, b \rightarrow b\},$$
$$h_4 = \{A \rightarrow a, B \rightarrow b, a \rightarrow a, b \rightarrow b\}.$$

Table h_1 introduces a symbol a in every derivation step and table h_3 introduces a symbol b in every derivation step. Table h_2 serves as switch between the symbols A and B and table h_4 terminates the derivation process. Consider the derivation for the word *baba*:

$$AA \underset{h_2}{\Longrightarrow} BB \underset{h_3}{\Longrightarrow} bBbB \underset{h_2}{\Longrightarrow} bAbA \underset{h_4}{\Longrightarrow} baba$$

The language generated is $L(G_2) = \{ww \mid w \in \{a,b\}^+\}$.

Example 3. Let $G_3 = (\{A,B,C,D,a,b,c,d\},\{h_1,h_2,h_3,h_4\},ABCD,\{a,b,c,d\})$ be an ET0L system, where the tables are given as follows:

$$h_1 = \{A \to aA, C \to cC, B \to B, D \to D, a \to a, b \to b, c \to c, d \to d\},$$
$$h_2 = \{A \to a, C \to c, B \to B, D \to D, a \to a, b \to b, c \to c, d \to d\},$$
$$h_3 = \{B \to bB, D \to dD, A \to A, C \to C, a \to a, b \to b, c \to c, d \to d\},$$
$$h_4 = \{B \to b, D \to d, A \to A, C \to C, a \to a, b \to b, c \to c, d \to d\}.$$

Using tables h_1 or h_3 introduce in every derivation step the symbols a and c or the symbols b and d, respectively. The tables h_2 and h_4 are used in order to terminate the derivation process. The language generated is $L(G_3) = \{a^n b^m c^n d^m \mid n,m \geq 1\}$.

3 L Systems as Bio-inspired Multi-agent Systems

Lindenmayer systems may be interpreted in terms of multi-agent systems. ET0L systems define systems of distributed components in which components can be viewed as autonomous problem solvers that must collaborate in order to perform complex tasks. Every table h in an L system can be seen as an agent that has a particular concern. Each agent has its rules and participate in the solution of a complex problem: the generation of the language.

Ferber [5] defines a multiagent system as having the following basic entities. Those basic entities can be interpreted in terms of L Systems as follows:

- An *environment*. In our system, the environment where the agents collaborate is the *string* they are rewriting (starting by the axiom).
- A *set of objects* that exist in the environment. The objects in the environment are *symbols* over the alphabet.
- A *set of agents*. The agents in ET0L systems are *the tables h in H*. Each table h is an agent that has a particular concern.
- A *set of operations* that agents can use to sense and affect objects. The set of operations in L systems are the *rules* each table h has that allow them to participate in the solution of the problem.
- A set of *universal laws* that define the reaction of the environment to agent operations. In our case, the reaction of the environment to agent operations is just the *modification of the string* due to the rules applied by the agents.

According to Wooldridge [19], agents in a multi-agent systems have three important characteristics:

- *Autonomy*: the agents are at least partially autonomous.
- *Local views*: no agent has a full global view of the system.
- *Decentralization*: there is no designated controlling agent.

In L Systems, agents h are *autonomous* since each of them has its set of rules that are independent on the rules other agents in the system have. Agents have local views, since no agent alone can solve the problem of generating the language of the system, but collaboration between all the agents in the system is needed in order to generate the language. And, of course, no agent in L systems controls the whole process.

From what we have said, it follows that L systems can be considered as multi-agent systems. Moreover, Lindenmayer systems can be seen as *bio-inspired* multi-agent systems. In fact, L systems are the first bio-inspired model in the field of formal language theory. Aristid Lindenmayer introduced L systems in 1968 as a theoretical framework in order to model the development of filamentous organisms, which are composed of cells. These cells receive inputs from their neighbours and change their states and produce outputs based on their states and the input received. Cell division is modelled by inserting two new cells in the filament in order to replace one cell. With these theoretical framework of Lindenmayer an organism as a whole was provided that models individual acts of division, unequal divisions, interaction of two or more cells and cell enlargement. The possible combinations of interactions among more than a handful of cells becomes rapidly unmanagable without a mathematical theory and computer application, which are provided both in the framework of Lindenmayer systems.

4 Applying L Systems to Natural Language Processing

> 'There are different ways to organize a generation system to gain more flexibility. The ideal organization lets each module contribute the choices to do with its area concern, without having to make other decisions that would be better made by some other module.' [1, p. 504].

The above quotation clearly defends as ideal organization in *natural language processing* the multi-agent one. To consider a distributed structure, where several agent components have limited information and operate in an independent way, may facilitate very much the task of processing natural language. In general, formal and computational approaches to natural language demand non-hierarchical, parallel, distributed models in order to explain the complexity of linguistic structures as the result of the interaction of a number of independent but cooperative components. In natural language processing, researchers have turned from the initial serial models to highly parallel ones. *Distribution* and *cooperation* are two important notions in natural language processing. In this field, it is usual to deal with complex tasks *distributed* among a set of 'processors' that *work together* in a well defined way. In a distributed/cooperative architecture for natural language processing, agents can work together to solve problems that are beyond their individual capabilities. Each agent in the system can work independently, but problems faced by each of them cannot be completed without cooperation. Cooperation is necessary because no agent has sufficient expertise, resources and information to solve the problem. The suitability shown by cooperative distributed architectures, have led to apply distributed models to the field of natural language processing. In text analysis, for example, we can find models as *TALISMAN* [17] defined as a distributed architecture for text analysis in French, that includes linguistic agents that correspond either to classical levels in linguistics (morphology, syntax, semantics) or to complex language phenomena. The main goal of this model is to show that complex linguistic phenomena can be defined and processed in a distributed architecture. The

CARAMEL model proposed in [7] is also an example of distributed architecture in natural language understanding. The model proposed by those authors intends to be applied to several different topics such as text understanding, dialogue management, making abstracts or intelligent tutoring systems.

The idea of distribution present in multi-agent systems is adequate for every component of a grammar. In [3], a highly distributed organization of *syntax* where components are determined by representations they recover, is suggested. The author defends four modules in the syntactic processor, each related with a 'representational' or 'informational' aspect of grammar: Phrase Structure; Chains; Thematic Structure; Coindexation (Binding and Control Theory). Another example of distribution, this time in *semantic and phonological* modules, is presented in [9] where the author suggests that: *'... meaning, like phonological structure, is organized into independent but interacting tiers, each of which contributes a different class of conceptual distinctions to meaning as a whole.'* [9, p. 2]. Studies on modularity in *morphology* can be found in [4], for example. About the internal modularity of *pragmatics* some approaches are presented in [18] and [8]. So, from all those examples it follows that not only a grammar as a whole can be regarded as a modular system composed by several parallel interacting components (syntax, semantics, phonology, etc.), but that also every component that builds up grammar can be viewed as *internally distributed*, being divided into several different agents as well.

From what we have said above, it follows that multi-agent systems offer the necessary flexibility in natural language processing. The aim of a multi-agent system for natural language processing is to define a set of agents with different skills that participate in processing of natural language. L systems can be seen as multi-agent systems that offer natural language processing a good tool to deal with the complex task of generate/understanding natural language.

Moreover, L systems present another interesting feature to be applied to natural language processing: *bio-inspiration*. The biological inspiration of L systems is an interesting property from a linguistic point of view since in the last decades there seems to be a tendency to use bio-inspired models in the description and processing of natural languages. Linguistics has not been able to solve the problem of generation/understanding natural language, partly because of the fail in the models adopted. Bio-inspired models could be a possible solution since one of the advantages of such kind of models is to offer more 'natural' tools than the ones used so far. Rewriting methods used in a large number of natural language approaches seem to be not very adequate, from a cognitive perspective, to account for the processing of language.

In the following we sketch one idea of how ET0L systems could be applied to deal with different linguistical levels. We consider the syntactical and the semantical level and thus an ET0L system with two tables h_{syn} and h_{sem}, where h_{syn} deals with syntax and h_{sem} with semantics. The table h_{syn} checks whether a given sentence is syntactically correct and the table h_{sem} assigns the semantical structure. The example is simplified in order to illustrate better the applicability of L systems for natural language processing. The number of the linguistical levels and the theories for these linguistical levels can be choosen as liked. Let h_{syn} be the set of rules

$\{S \rightarrow N_p\ V_p, N_p \rightarrow Susan, V_p \rightarrow V\ N_p, Agent \rightarrow Agent, V \rightarrow kissed, N_p \rightarrow John\}$ and h_{sem} be the set of rules $\{Susan \rightarrow Agent, V \rightarrow V, N_p \rightarrow N_p, kissed \rightarrow Action, John \rightarrow Patient, Agent \rightarrow Agent\}$. Let S be the axiom. Table h_{syn} rewrites the axiom to $N_p\ V_p$. Now again only table h_{syn} is applicable and rewrites all symbols (namely $N_p\ V_p$) to $Susan\ V\ N_p$. At this stage table h_{syn} is no more applicable but table h_{sem} is and rewrites all symbols into $Agent\ V\ N_p$. Applying table h_{syn} again yields $Agent\ kissed\ John$. Finally table h_{sem} is applied and rewrites all symbols into $Agent\ Action\ Patient$. In this simplified example one table assigns a syntactical structure to a given sentence and as soon as a semantical category can be assigned to a syntactical category this is done by the other table.

5 Conclusions

Formal and computational approaches of natural language demand non-hierarchical, parallel, distributed models in order to explain the complexity of linguistic structures as the result of the interaction of a number of independent but cooperative modules. Natural language processing needs frameworks that are able to generate/understand the structures present in natural languages with simple mechanisms that describe/explain those structures in a more natural way than the usual rewriting systems. In natural language processing, researchers have turned from the initial serial models to highly parallel ones. In theoretical linguistics, the hierarchical view of grammar has revealed as problematic and there has been a search for systems with parallel and autonomous components that cooperate in order to generate natural language. Since languages, either natural or artificial, are particular cases of symbol systems and the manipulation of symbols is the stem of formal language theory, it seems adequate to look for bio-inspired models that have been defined in that research area. Lindenmayer systems are the first bio-inspired model proposed in the field of formal languages and it is also the first one that replaces the sequential rewriting by the parallel one. Therefore, it seems that Lindenmayer systems offer a great deal of the properties that seem to be necessary in order to approach linguistic structures. Moreover, it has been pointed out that using a multi-agent system for natural language processing has several advantages and, as we have showed in this paper, L systems can be considered as multi-agent systems. Models in natural language processing demand *bio-inspired devices* that avoid the *sequential* rewriting, that have *enough expressiveness* to describe natural languages and that present a *distributed* architecture. L systems are described as a *parallel* and non sequential grammatical formalism, they are *biologically* inspired, they can generate the *non-context free structures* present in natural language and can offer a distributed/cooperative architecture. Therefore, we consider that it could be interesting to apply L systems to the description/processing of natural language in order to see if this parallel distributed bio-inspired model may improve current NLP approaches.

Acknowledgements. The research of Leonor Becerra-Bonache was supported by a Marie Curie International Fellowship within the 6^{th} European Community Framework Programme.

References

1. Allen, J.F.: Natural Language Understanding. The Benjamin/Cummings Publishing Company, California (1987)
2. Aref, M.M.: A Multi-Agent System for Natural Language Understanding. In: KIMAS 2003, October 1-3 (2003)
3. Crocker, M.W.: Multiple Meta-Interpreters in a Logical Model of Sentence Processing. In: Brown, C.G., Koch, G. (eds.) Natural Language Understanding and Logic Programming, III, pp. 127–145. Elsevier Science Publishers B.V., North-Holland (1991)
4. Everaert, M., Evers, A., Hybreqts, R., Trommelent, M. (eds.): Morphology and Modularity: In Honour of Henk Schultink. Publications in Language Sciences 29, Foris (1988)
5. Ferber, J.: Multi-Agent Systems. An Introduction to Distributed Artificial Intelligence. Addison-Wesley, Harlow (1999)
6. Fum, D., Guida, G., Tasso, C.: A Distributed Multi-Agent Architecture for Natural Language Processing. In: Proceedings of COLING 1988, Budapest, vol. II, pp. 22–27 (1988)
7. Grau, B., Sabah, G., Vilnat, A.: Control in Man-Machine Dialogue. Think 3, 32–55 (1994)
8. Horn, L.R.: Pragmatic Theory. In: Newmeyer, F. (ed.) Linguistic Theory: Foundations, Linguistics: The Cambridge Survey, vol. I, pp. 113–145. Cambridge University Press, Cambridge (1988)
9. Jackendoff, R.: Semantic Structures. MIT Press, Cambridge (1990)
10. Jiménez-López, M.D.: A Grammar Systems Approach to Natural Language Grammar. Linguistics and Philosophy 29, 419–454 (2006)
11. Kari, L., Rozenberg, G., Salomaa, A.: L systems. In: Rozenberg, G., Salomaa, A. (eds.) Handbook of Formal Languages, vol. 1. Springer, Berlin (1997)
12. Lima, V.L.S., Abrahao, P.R.C., Santa Maria Filho, I.: Distributing linguistic knowledge in a multi-agent natural language processing system: re-modelling the dictionary. Procesamiento del lenguaje natural 23, 104–109 (1998)
13. Lindenmayer, A.: Mathematical models for cellular interaction in development. Journal of Theoretical Biology 18, 280–315 (1968)
14. Rozenberg, G., Salomaa, A.: The Mathematical Theory of L-Systems. Academic Press, New York (1980)
15. Rozenberg, G., Salomaa, A.: Handbook of Formal Languages. Springer, Berlin (1997)
16. Silva, J.L., Lima, V.L.S.: An alternative approach to lexical categorial disambiguation using a multi-agent systems architecture. In: Proceedings of the RANLP 1997, Bulgaria, pp. 6–12 (1997)
17. Stefanini, M.H., Berrendoner, A., Lallich, G., Oquendo, F.: TALISMAN: un système multi-agents gouverné par des lois linguistiques pour le traitement de la langue naturelle. In: Proceedings of the COLING 1992, Nantes, pp. 23–28 (1992)
18. Wilson, D., Sperber, D.: Pragmatics and Modularity. In: Davis, S. (ed.) Pragmatics. A Reader, pp. 583–594. Oxford University Press, New York (1991)
19. Wooldridge, M.: An Introduction to Multi Agent Systems. John Wiley & Sons Ltd., New York (2002)

Poetic RNA: Adapting RNA Design Methods to the Analysis of Poetry

Veronica Dahl, M. Dolores Jiménez-López, and Olivier Perriquet

Abstract. The style in which a Ribonucleic Acid (RNA) molecule folds in space obeys laws of nucleotide binding and attraction which are encoded in its primary structure, that is, in the sequence of nucleotides conforming it. Natural language sentences can also be viewed as encodings for a structure in space -a parse tree- which exhibits relationships or bindings between its parts. We explore the possibilities in adapting a recent and simple methodology for bioinformatics -which has been successfully used for RNA design- to the problem of parsing poems that follow specific stylistic trends. The methodology introduced in this paper can be expressed in terms of a multi-agent system that includes two types of agents: linguistic agents and probabilistic agents.

1 Introduction

The application to molecular biology of AI methods such as logic programming and constraint reasoning constitutes a fascinating interdisciplinary field which, despite being relatively new, has already proved quite fertile.

As well, methodologies that pertain to the natural language processing field of AI are now being exploited to analyze biological sequences, which is uncovering

Veronica Dahl
Department of Computing Science, Simon Fraser University, Burnaby, B.C., Canada
Research Group on Mathematical Linguistics, Universitat Rovira i Virgili,
43002 Tarragona, Spain
e-mail: veronica@cs.sfu.ca

M. Dolores Jiménez-López
Research Group on Mathematical Linguistics, Universitat Rovira i Virgili,
43002 Tarragona, Spain
e-mail: mariadolores.jimenez@urv.cat

Olivier Perriquet
Centre for Artificial Intelligence, New University of Lisbon, 2829-516 Caparica, Portugal
e-mail: olivier@perriquet.net

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 403–410.
springerlink.com

similarities between the languages of molecular biology and human languages [3]. On the other hand, the potential influence of computational molecular biology over natural language processing, has been much less studied, although some similarities and cross-fertilization potential are starting to be identified and exploited [5].

In [2], a novel methodology was presented -chrRNA (Constraint Handling Rules RNA)- for addressing an important bioinformatics problem which has been proved to be computationally hard: that of finding an RNA sequence which folds into a given structure. Obvious potential applications are to in vitro genetics, by enabling the scientists to produce RNAs artificially from sequences; and to drug design, which typically progresses backwards from proteins to RNAs and finally to DNAs.

A much less evident potential application is proposed in this paper: the adaptation of the same methodology to computational linguistics problems. We explore in particular its uses for solving the problem of using stylistic information as an aid for processing poetry. Our work springs from the observation that, in the same way as molecules each have their own style, while all obeying the same nucleic acid grammar (that of RNA or DNA), human artistic creations are marked by the specific style of their author even when the grammar available to all authors may be the same.

The method developed in [2] for determining what RNA sequence a given molecule's structure folds into involves a very simple grammar augmented by probabilities. These probabilities encode the molecule's "style", as it were, so adapting our method to computational linguistics involves resorting to stylistic probabilities observed in a given author's poetic production in order to aid in the parse of a given poem of the same author, or to aid in determining authorship itself.

2 Background

2.1 Constraint Handling Rules (CHR) and Grammars (CHRG)

Constraint Handling Rules (CHR) provide a simple bottom-up framework which has been proved to be useful for algorithms dealing with constraints [6, 4]. Because logic terms are used, grammars can be described in human-like terms and are powerfully extended through (hidden) logical inference. The format of CHR rules is: `Head==>Guard|Body`. `Head` and `Body` are conjunctions of atoms and `Guard` is a test constructed from (Prolog) built-in or system-defined predicates. The variables in `Guard` and `Body` occur also in `Head`. If the `Guard` is the constant "true", then it is omitted together with the vertical bar. Its logical meaning is the formula $(Guard \rightarrow (Head \rightarrow Body))$ and the meaning of a program is given by conjunction. There are three types of CHR rules: (1) *Propagation rules*, which add new constraints (body) to the constraint set; (2) *Simplification rules*, which also add as new constraints those in the body, but remove as well the ones in the head of the rule. (3) *Simpagation rules*, which combine propagation and simplification traits, and allow us to select which of the constraints mentioned in the head of the rule should remain and which should be removed from the constraint set.

The rewrite symbols for the first two rules are respectively: `==>`, `<=>` and for simpagation rules, the notation is `Head1\Head2<=>body`. Anything in `Head1` remains in the constraint set and anything in `Head2` is removed from the constraint set.

CHR have a grammatical counterpart -CHRG [4]- which is to CHR what Definite Clause Grammars are to Prolog: grammar symbols compile into constraints in which the word boundaries are made explicit automatically, so there's no need to handle them in the grammar. The notation that distinguishes a grammar rule from a CHR rule is `::>`, e.g. the CHRG rule: `verb, noun_phrase::> verb_phrase` compiles into a CHR counterpart in which the start and end points are visible:

```
verb(Start,P1),
noun_phrase(P1,End)==>verb_phrase(Start,End).
```

2.2 The chrRNA Method for RNA Design

Research on the language of nucleic acid, parsimoniously but powerfully formed basically with only four letters, or nucleotides (A, C, T and G), has in the past few years started to uncover fruitful similarities between this language and human languages [10]. Just as a sentence in English is only the visible part, or tip of the iceberg, for that sentence's rich structure, which is essential to its meaning, a nucleic acid "sentence" (i.e., a sequence of A's, C's, T's and G's) hides, or codes for, an incredibly rich structure which takes 3D shape when some of the nucleotides in the sentence bind with others, and which is just as essential to that sentence's meaning.

Scientists in computational molecular biology are interested both in the generation aspect of this process, namely predicting which 3D structure a known sequence of nucleotides might fold into, and in its analyzing aspect, namely finding out what sequence of nucleotides codes for a molecule whose 3D structure is known.

Both these aspects have approximate computational solutions, as long as compromises are made with a particularly hard subproblem: that of molecules containing a specific 3D structure called a *pseudoknot*. Simple pseudoknots occur when a structure of the form of a hairpin loop contains nucleotides that bind outside the loop.

Methods dealing with pseudoknots are delicate to conceive. Initially, the methods for RNA structure prediction used to simply disregard the possibility of having pseudoknots, due to the huge algorithmic complexity these would bring to the problem, compared to the moderate gain in prediction accuracy. The solution proposed in [2] allows pseudoknots. Bavarian and Dahl show that by using a set of simple Constraint Handling Rules, this problem could actually have an approximate but somehow useful solution in linear time, despite the simplistic grammar model. This solution- named chrRNA- encodes an RNA molecule's style in terms of probabilities which are incorporated into the (otherwise highly ambiguous) grammar rules that describe RNA primary structure. With the aid of their embedded probabilities, they guide the assignment of content (one of A, C, U[1] or G) to nucleotides we know

[1] In RNA, thymine (T) is replaced by uracile (U), but U and T are perfectly equivalent at the informational level.

are paired but whose identity is not known, and also for the nucleotides we know are unpaired but whose identity is likewise unknown.

These rules are based on the highly ambiguous context free grammar proposed in [1] for RNA secondary structure prediction: $[S \rightarrow cSg|gSc|aSu|uSa|gSu|uSg; S \rightarrow aS|gS|uS|cS; S \rightarrow a|g|u|c; S \rightarrow SS$. Using CHR to implement this grammar allows us both to exploit the bottom-up characteristic of CHR rules, and to keep track of ambiguous readings with no special overhead.

2.2.1 Representation Issues

As mentioned, the problem consists of finding a sequence of A, C, U and G which folds into the (given as input) structure of the desired RNA. In the chrRNA method, this structure is entered in the format of CHR constraints, e.g. for expressing that the base number 1 and the base number 43 in the sequence are paired together, we add the constraint `pair(1,43)` or if base number 3 is unpaired, the corresponding constraint would be `upair(3)`. One advantage of this input format to the input format used by previous methods (RNAinverse and RNA-SSD) is that it is capable of accepting pseudoknots in the input structure.

2.2.2 Processing Issues

We now need to assign nucleotides to each position given the input constraints. The trivial solution of randomly assigning one of the Watson-Crick pairs (an AU or CG pair) to each base pair and one of the four nucleotides (A, C, G and U) to the unpaired bases is not a reliable one, since we might end up with a sequence that may not actually fold into the input structure. This is because the number of GC pairs has an important role in stabilizing a certain structure. For instance, if we assign base G to 1 and base U to position 43 and if we have a base C in position 42, in the end, the structure might be `pair(1,42)` instead of `pair(1,43)`.

The solution proposed in [2] uses CHR rules combined with the probabilities that are believed to govern the proportion of base pairs within RNA sequences, calculated by comparing several RNAs together from Gutell lab's comparative RNA website, a database of known RNA secondary structures. After comparing 100 test cases with various length from 100 to 1500 bases, they found the following probabilities for each base pair: $P_{CG} = 0.53, P_{AU} = 0.35, P_{GU} = 0.12$ The other probabilities which are of interest are the probabilities for an unpaired base to be one of A, C, G, or U. The results are as follows: $P_G = 0.18, P_A = 0.34, P_C = 0.27, P_U = 0.20$.

Inserting the probabilities into the rules is done by generating a random variable in the guard section of the rules, and then testing this random variable according to the probabilities: for instance if the random variable `I` generated when trying to instantiate a pair of nucleotides X1 and Y1 is less than 0.53, the grammar rule will assign a GC pair to `pair(X1,Y1)`.

The result is a quite simple while powerful system which performs better than the two previous methods for longer sequences, and which is capable of handling pseudoknots. The application of their ideas could be useful for in vitro genetics and for drug design. We shall next argue that they can also be fruitfully adapted for literary style processing, and exemplify this thesis around the poetry of the Cuban author Nicolas Guillen.

3 Poetic RNA: A Multi-Agent Model

3.1 Poetic Style

Just as the CF grammar of RNA by itself is highly ambiguous, human languages in general and poems in particular are also prone to ambiguity; however we have chosen poems as a starting point because poetic style often departs in idiosyncratic ways from standard word orderings often found in prose, and identifying such style trends might therefore contribute clues that serve to disambiguate, much in the way as the probabilities used in chrRNA help identify an RNA string unambiguously. We can moreover express these trends precisely in the same way: through probabilities resulting from an analysis of the particular author's style.

As an example, Guillen's poems make frequent use of topicalization, which often makes it hard to identify a sentence's subject from syntax alone.

However, an analysis of Guillen's style might allow us to encode into the context free rules, just as we did for nucleic acid rules, probabilities that can help us disambiguate, e.g. by helping us determine in which cases an NP following a verb is its subject and in which cases it is an object. We might want to encode Spanish and stylistic observations such as: (1) a verb's subject has high probability of being adjacent to the main verb; (2) a direct object has high probability of following the verb; (3) a main verb which is repeated has high probability of having in its repeated occurrence the same subject as the first occurrence's; (4) a second verb with no overt subject has high probability of implicitly having the same subject as the first verb.

Of course, the more accurate an analysis of a given poet's style and of the exact values of the different probabilities, the more accurate the proposed method will be for correctly using style to interpret a poem's meaning. It is not our purpose in this paper to propose precise values for any such probabilities; rather, we aim at showing how the chrRNA method that has proved so successful for RNA design can be adapted also to literary analysis.

We now abstract the problem of analyzing poems to make it amenable to the same form as that of RNA design.

3.2 Agents in Poetic RNA

CHRG, in which our proposal is based, propitiate the introduction of agents since they allow multi-headed rules to drive the parsing process. If all heads in a rule have matching counterparts within the working store, then the rule applies. Therefore, a CHRG rule can deal for instance with both syntactic and semantic information in one stroke, by having syntactic and semantic agents coexisting as heads in the same rule.

Our model consists basically of a CHRG that incorporates two types of agents: a *linguistic agent* whose task is gleaning the basic syntactic structures and a *probabilistic agent* that encodes probability values that allow us to choose the more likely parsing in the case of ambiguity. In what follows we describe those two agents.

3.2.1 Linguistic Agent

The linguistic agent may be seen as the basic parser whose task consists in gleaning the basic syntactic structures. Notice that our "nucleotides" are now blocks of words marked by syntactic functions which can in a first stage be gleaned through a regular CHRG. For example, "el borracho" in the sentence "Dejó el borracho en su coche, dejó el cabaret" ("The drunkard left the cabaret... in his car") can be analyzed into a noun phrase (NP) stretching from position 2 in the input sentence to position 3 but whose role inside the sentence -the style which binds concepts with one another- remains to be found. This shows up in a CHR rule as the constraint `noun_phrase(2,3,[el,borracho])`, and in its CHRG counterpart, as `noun_phrase([el,borracho])`.

The task of our linguistic agent is to represent the roles we are after (subject, direct object...) through grammar symbols that compile into constraints. The roles will be superimposed over the stretch of input string concerned. Thus if we view NP as a label covering the substring "el borracho", once we find that this NP's role is that of subject, we will add a new constraint which, in this same graphic view, would label the same substring with "subject". Other arguments relevant to the analysis can of course be included, for our purposes here we'll only add in some cases an index I which will be an identifier to the "nucleotide" that the block containing it attaches to, or refers to.

3.2.2 Probabilistic Agent

Let us again consider the Guillen's sentence "Dejó el borracho en su coche, dejó el cabaret sombrio". This can be parsed into one sentence "The drunkard left the sombre cabaret (by traveling) in his car". Any Spanish reader would understand that the verb's repetition is for poetic effect, rather than a "new" main verb. A machine analyzer, however, would recognize two sentences, corresponding either to: (1) "(Someone) left the drunkard in his car, and (the same person) left the sombre cabaret, or (2) "Someone left the drunkard in his car, and the sombre cabaret left". The first interpretation is likely when one considers that implicit subjects are very common in Spanish; the second one is nonsensical for humans but plausible from syntax alone, and therefore, a fair candidate for a poetically uninformed parser.

In order to solve these cases of ambiguity, we add a *probabilistic agent*. In the case of RNA design the probabilistic agent operates in the guard of a CHR rule, and uses the probabilities that are believed to govern the proportion of base pairs withing RNA sequences [2]. In our Poetic RNA, the probabilistic agent encodes probability values that will allow us to determine, in case of ambiguity, which possible analysis is more likely. Thus if we've encoded probabilities to the effect that when a verb is not the initial word in a sentence, the NP that follows it is likely to be a direct object rather than a subject, the first rule below will capture that NP into the verb phrase (by creating a `verb_phrase` symbol that spans both substrings, i.e. from P1 to P3), while marking it as the direct object of that verb (by associating the same index, J, to both the verb and the object). This rule will be used for instance for "dejó el cabaret sombrío". Notice that we express it as a CHR rule in order to access the word boundaries and generate the new symbols with appropriate span.

If our probabilities moreover indicate that initial verbs in Guillen's poetry are likely to appear before the subject NP just for stylistic effect, the second of the following two rules will be taken for "dejó el borracho". This rule records the role of subject found for this NP and marks both it and the verb with an index I, to indicate that the string L2 is the subject of the verb L1. Notice that since we don't need word boundaries made explicit, this rule is a CHRG one- the word boundaries will be added automatically at compilation time.

```
verb(P1,P2,L1),
noun_phrase(P2,P3,L2)==>prob(subj_verb_inversion,low),
append(L1,L2,L)|verb(p1,p2,L1,J),
direct_object(P2,P3,L2,J),verb_phrase(P1,P3,L).
verb(L1),
noun_phrase(L2)::>prob(subj_verb_inversion,high)|
verb(L1,I),subject(L2,I)—
```

Likewise, the probabilistic agent could encode probabilities to express the likelihood that a repeated verb's implicit subject refers to the subject of that same verb's other occurrence, and consult them in the guard of the concerned rule in order to co-index both occurrences of that verb with the same subject.

4 Concluding Remarks

Algorithmic approaches to poetry have been around for a relatively long time. They mostly focus on generating poetry by automated or semi-automated means [9], and as such, belong to the general field of Electronic Writing. Automated poetry analysis, on the other hand, remains a bit more elusive, and concentrates on the more mechanizable subtasks, such as automated analysis of sound and meter [8].

To the best of our knowledge, this work is the first in approaching poetic analysis through stylistic probabilities added to a constraint-based parser. The exploitable similarities between molecular and poetic style are potentially much bigger than explored in this first paper on the subject, and relatively straightforward to implement in the underlying model we propose. What we just exemplified with a process called topicalization, would gain in being pursued on a corpus of known poetic techniques that make use of parsing ambiguity. In the scope of this paper, note that the ground level of our analysis is syntactic, in that respect we do not consider for instance what may be termed recombinant words or concepts [11] that would be visible only at an infra-syntactic level. These remarks give a picture of the horizon we assign to our work.

Searls exposes in [10] the important role of computational linguistics techniques in the domain of nucleic acid string analysis. Based on the indubitable efficiency of these techniques in bioinformatics, he terms the processes at work in the production of protein "the language of the gene" and "the language of proteins". The metaphor made its own way and the field of linguistics is currently well anchored in the domain. The linguistic denomination is more than a mere metaphor due to the concrete operability of linguistics tools. When speaking about a molecule's style, we actually go one step further in that anthropomorphic projection, and we do it on purpose, in a very prospective manner. Our intention is to to explore a backward genre contamination where the linguistic approach itself gets inspired by its own application to nucleic acid analysis.

References

1. Baldi, P.: Bioinformatics: the machine learning approach. MIT Press, Cambridge (1998)
2. Bavarian, M., Dahl, V.: Constraint-Based Methods for Biological Sequence Analysis. Journal of Universal Computing Science 12(11), 1500–1520 (2006)
3. Bel-Enguix, G., Dahl, V., Jiménez-López, M.D.: DNA and Natural Languages: Text Mining. In: Fred, A. (ed.) Proceedings of the International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, pp. 140–145. INSTICC, Madeira (2009)
4. Christiansen, H.: CHR as Grammar Formalism, a First Report. In: Apt, K.R., et al. (eds.) Sixth Annual Workshop of the ERCIM Working Group on Constraints, Prague (2001)
5. Dahl, V., Maharshak, E.: DNA Replication as a Model for Computational Linguistics. In: Mira, J., Ferrández, J.M., Álvarez, J.R., de la Paz, F., Toledo, F.J., et al. (eds.) IWINAC 2009. LNCS, vol. 5601, pp. 346–355. Springer, Heidelberg (2009)
6. Fruhwirth, T.: Theory and Practice of Constraint Handling Rules. Journal of Logic Programming 37(1-3), 95–138 (1998)
7. Guillen, N.: Motivos de son (1930); Biblioteca Virtual Miguel de Cervantes, Alicante (2001)
8. Logan, H.M.: Computer Analysis of Sound and Meter in Poetry. College Literature 15(1), 19–24 (1988)
9. Mendelowitz, E.: Drafting poems: inverted potentialities. In: International Multimedia Conference, Proceedings of the 14th Annual ACM International Conference on Multimedia, Santa Barbara, pp. 1047–1048 (2006)
10. Searls, D.B.: The computational linguistics of biological sequences. In: Artificial intelligence and molecular biology. American Association for Artificial Intelligence, pp. 47–120 (1993)
11. Seaman, W.: Recombinant Poetics: Emergent Meaning as Examined and Explored Within a Specific Generative Virtual Environment. Ph.D. Thesis, CAiiA (1999)

A Visual Language for Modelling and Simulation of Networks of Evolutionary Processors

Antonio Jimenez, Emilio del Rosal, and Juan de Lara

Abstract. The goal of this work is to provide the jNEP user with a visual tool to graphically design the NEPs under consideration. jNEP is a Java simulator for NEPs previously developed by some of the authors of this paper. We have designed a domain specific visual language for NEPs by means of $AToM^3$. We have also taken advantage of the $AToM^3$'s graph grammar modules to automate some mechanical and time-consuming designing tasks, such as properly placing filters close to their processors, and defining some kinds of standard graph topologies. $AToM^3$ is a python tool previously developed by some of the authors of this paper.

1 Introduction

1.1 Introduction to $AToM^3$

Visual Languages play a central role in many computer science activities. For example, in software engineering, diagrams are widely used in most of the phases of software construction. They provide intuitive and powerful domain-specific constructs and allow the abstraction from low-level, *accidental* details, enabling reasoning and improving understandability and maintenance. The term Domain Specific Visual Language (DSVL) [1] refers to languages that are especially oriented to a certain domain, limited but extremely efficient for the task to be performed. DSVLs are extensively used in Model Driven Development, one of the current approaches to Software Engineering. In this way, engineers no longer have to resort to low-level languages and programming, but are able to synthesize code for the final application from high-level, visual models. This increases productivity, quality, and permits its use by non-programmers.

Antonio Jimenez · Emilio del Rosal · Juan de Lara
Escuela Politécnica Superior - Universidad Autónoma de Madrid, Campus de Cantoblanco 28049, Madrid, Spain
e-mail: antonio.jimenez@uam.es, emilio.delrosal@uam.es, juan.delara@uam.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 411–418.
springerlink.com

The design of a DSVL involves defining its concepts and the relations between them. This is called the abstract syntax, and is usually defined through a meta-model. Meta-models are normally described through UML class diagrams. Hence, the language spawned by the meta-model is the (possibly infinite) set of models conformant to it. In addition, a DSVL needs to be provided with a concrete syntax. That is, a visualization of the concepts defined in the meta-model. In the most simple case, the concrete syntax just assigns icons to meta-model classes and arrows to associations. The description of the abstract and concrete syntax is enough to generate a graphical modelling environment for the DSVL. Many tools are available that automate such task, in this paper we use AToM^3 [2].

In many scenarios, the description of the DSVL syntax is not enough, but one would like to define manipulations that "*breath life*" into such models. For example, one could like to animate or simulate the models, to define "macros" for complex editing commands, or build code generators for further processing by other tools. As models and meta-models can be described as attributed, typed graphs, they can be visually manipulated by means of graph transformation techniques [3]. This is a declarative, visual and formal approach to manipulate graphs. Its formal basis, developed in the last 30 years, makes possible to demonstrate properties of the transformations. A graph grammar is made of a set of rules and a starting graph. Graph grammar rules are made of a left and a right hand side (LHS and RHS), each one having graphs. When applying a rule to a graph (called host graph), an occurrence of the LHS should be found in the graph, and then it can be replaced by the RHS.

In this paper, we apply the aforementioned concepts to build a DSVL to design Networks of Evolving Provessors (NEPs). For this purpose, we built a meta-model in the AToM^3 tool and a graphical modelling environment was automatically generated. Then, such environment was enriched by providing rules to automate complex editing commands, and by a code generator to synthesize code for jNEPs, in order to perform simulations. The approach has the advantage that the final user does not need to be proficient in the jNEPs textual input language, but he can model and simulate NEPs visually.

1.2 Introduction to NEPs

Networks of evolutionary processors (NEPs) are a new computing mechanism directly inspired in the behaviour of cell populations. Each cell contains its own genetic information (represented by a set of strings of symbols) that is changed by some *evolutive* transformations (implemented as elemental operations on strings). Cells are interconnected and can exchange information (strings) with other cells. NEPs were initially used as generating devices in [4], [5], and [6].

The Connection Machine [7], the Logic Flow paradigm [8] and the biological background of DNA computing [9], membrane computing [10], and specially the theory of grammar systems [11] can be considered precedents to NEPs.

A NEP can be defined as a graph whose nodes are processors which perform very simple operations on strings and send the resulting strings to other nodes. Every node has filters that block some strings from being sent and/or received.

1.2.1 NEPs: Definitions and Key Features

Following [12] we introduce the basic definition of NEPs.

Definition 1. A Network of Evolutionary Processors of size n is a construct:

$$\Gamma = (V, N_1, N_2, ..., N_n, G),$$

where:

- V is an alphabet and for each $1 \leq i \leq n$,
- $N_i = (M_i, A_i, PI_i, PO_i)$ is the i-th evolutionary node processor of the network. The parameters of every processor are:
  - M_i is a finite set of evolution rules of just one of the following forms:
    i. $a \to b$, where $a, b \in V$ (substitution rules),
    ii. $a \to \varepsilon$, where $a \in V$ (deletion rules),
    iii. $\varepsilon \to a$, where $a \in V$ (insertion rules).
  - A_i is a finite set of strings over V. The set A_i is the set of initial strings in the i-th node.
  - PI_i and PO_i are subsets of V^* respectively representing the input and the output filters. These filters are defined by the membership condition, namely a string $w \in V^*$ can pass the input filter (the output filter) if $w \in PI_i (w \in PO_i)$. In this paper we will use two kind of filters:
    · Those defined as two components (P, F) of *P*ermitting and *F*orbidding contexts (a word w passes the filter if $(\alpha(w) \subseteq P) \wedge (F \cap \alpha(w) = \Phi)$).
    · Those defined as regular expressions r (a word w passes the filter if $w \in L(r)$, where $L(r)$ stands for the language defined by the regular expression r).
- $G = (\{N_1, N_2, \ldots, N_n\}, E)$ is an undirected graph called the underlying graph of the network. The edges of G, that is the elements of E, are given in the form of sets of two nodes. The complete graph with n vertices is denoted by K_n.

A configuration of a NEP is an n-tuple $C = (L_1, L_2, \ldots, L_n)$, with $L_i \subseteq V^*$ for all $1 \leq i \leq n$. It represents the sets of strings which are present in any node at a given moment.

A given configuration of a NEP can change either by an *evolutionary* step or by a *communicating* step. When changing by an evolutionary step, each component L_i of the configuration is changed in accordance with the evolutionary rules associated with the node i. The change in the configuration by an evolutionary step is written as $C_1 \Rightarrow C_2$.

When changing by means of a communication step, each node processor N_i sends all the copies of the strings it has, able to pass its output filter, to all the node processors connected to N_i, and receives all the copies of the strings sent by any node processor connected with N_i, if they can pass its input filter. The change in the configuration by means of a communication step is written as $C_1 \vdash C_2$.

1.3 Introduction to jNEP and jNEPview

The jNEP [13] Java program, freely available at http://jnep.edelrosal.net, can simulate almost every type of NEP in the literature. The software has been developed under the following main principles: 1) it rigorously complies with the formal definitions found in the literature; 2) it serves as a general tool, by allowing the use of the different NEP variants and can easily be adapted to possible future extensions of the NEP concept.

jNEP consists of three main classes (*NEP*, *EvoluionaryProcessor* and *Word*), and three Java interfaces (StoppingCondition, *Filter* and *EvolutionaryRule*) This design mimics the NEP model definition. In jNEP, a NEP is composed of evolutionary processors, stopping conditions and an underlying graph (attribute *edges*), used to define the net topology. Likewise, an evolutionary processor contains a set of rules and filters.

jNEP reads the definition of the NEP from an XML configuration file that contains special tags for any relevant components in the NEP (alphabet, stopping conditions, the complete graph, every edge, the evolutionary processors with their respective rules, filters and initial contents). Despite the complexity of these XML files, the interested reader can see that the tags and their attributes have self-explaining names and values.

We show below, as an example, the configuration file of a very simple NEP. It has two nodes that, respectively, delete and insert the symbol *B*. The initial word *AB* travels from one node to the other. The first node removes the symbol *B* from the string before leaving it in the net. The other node receives the string *A* and adds again the symbol *B*. The resulting string comes back to the initial node and the same process takes place again.

```
<NEP nodes="2">
  <ALPHABET symbols="A_B"/>
  <GRAPH>
    <EDGE vertex1="0" vertex2="1"/>
  </GRAPH>
  <EVOLUTIONARY_PROCESSORS>
    <NODE initCond="A_B">
      <EVOLUTIONARY_RULES>
        <RULE ruleType="deletion" actionType="RIGHT" symbol="B" newSymbol=""/>
      </EVOLUTIONARY_RULES>
      <FILTERS>
        <INPUT type="2" permittingContext="A_B" forbiddingContext=""/>
        <OUTPUT type="2" permittingContext="A_B" forbiddingContext=""/>
      </FILTERS>
    </NODE>
    <NODE initCond="">
      <EVOLUTIONARY_RULES>
        <RULE ruleType="insertion" actionType="RIGHT" symbol="B" newSymbol=""/>
      </EVOLUTIONARY_RULES>
      <FILTERS>
        <INPUT type="2" permittingContext="A_B" forbiddingContext=""/>
```

```
        <OUTPUT type="2" permittingContext="A_B" forbiddingContext=""/>
      </FILTERS>
    </NODE>
  </EVOLUTIONARY_PROCESSORS>
  <STOPPING_CONDITION>
    <CONDITION type="MaximumStepsStoppingCondition" maximum="8"/>
  </STOPPING_CONDITION>
</NEP>
```

jNEP produces a sequence of log files, one for each simulation step. These logs contain a line for each processor in the same implicit order in which they appear in the configuration file. Each line contains the strings of the corresponding processor.

Besides, a graphical trace viewer of the NEP simulation has been developed. It was designed to be integrated in jNEP and its name is jNEPview [14] (freely available at http://jnepview.e-delrosal.net).

2 NEPs Visual Language

The system consists of four parts: two are core and essential parts of it (**meta-model**, which gives the elements that will be used to build models; and the **code generator**, which is a program that creates the code to be used by the simulator), and the other two are just helpful in giving usability (**constraints**, which are defined inside the meta-model and control some aspects of the models to ensure that they are syntactically correct; **graph grammars** are used to generate parts of the models crated that might be dull to be done manually).

This way the user perceives that only a few buttons, GUI elements, and common actions are needed to build a model that can be executed by the simulator, not being aware of the complexity of the code used to feed it.

From this point the parts which compose the system are described.

Fig. 1 presents the **UML class diagram** of the meta-model that represents the NEPs domain for the simulator. We can see several classes for the usual elements of a NEPs system in it: alphabet, processors, filters, rules, and stopping conditions. We need the following subclasses:

- rule → inserting, deleting, deriving, substituting, and regular_expression
- stopping_condition → consecutive_config, maximum_steps, words_disappear, and non_empty_node

Code generator is the other core part of the system. It is responsible of creating the XML file that will feed the simulator (using Python code). In order to do this, it performs couple of tasks: first of all it checks some properties that the model must meet, and then it goes through the instances that compose the model created by the user in order to generate parts of the XML.

In the first task the properties that are checked are the following: an alphabet must exist, there must be one stopping condition, there cannot be symbols in the model which are not in the alphabet, and there are not more than one connection from one processor to another one. Some of these properties are checked in the time of model creation by using constraints, but we check them here to increase reliability.

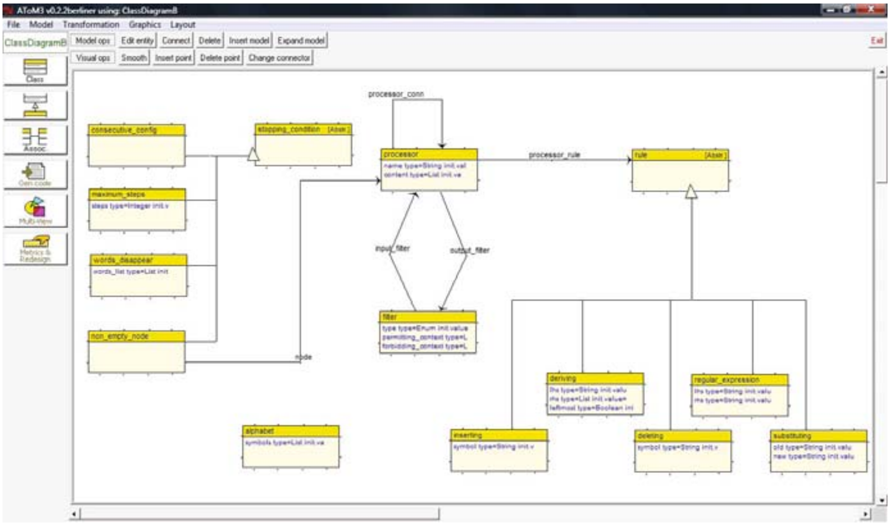


Fig. 1 The meta-model UML class diagram

After checking the model is valid, the second task is executed, where a loop over the instances is performed and, using their connections and attributes values, the XML file is written.

Graph grammars formal tool is used here to maximize the usability and make model design less dull and repetitive to the user.

The problems solved by means of this tool are the automatic assignment of input and output filters to processors, and automatic creation of connections among processors in order to get a fully connected graph.

Even being related to the final user's point of view, the correspondence among classes (non abstract) and graphical elements is an important part of the design phase. These direct relations are: alphabet → big rectangle, processor → small rectangle, filter → triangles, rule's subclasses → ovals, stopping_condition's subclasses → A text containing the name of the type of stopping condition.

All of them show its attributes' values as well.

2.1 User Point of View - How Domain Experts Use It

The way of making models is very simple: using GUI elements (like buttons) and dropping entities on a canvas. Also other well-known elements like combo-boxes or text input areas might be used.

Then, we can imagine that AToM^3 interface is quite intuitive. In Fig. 2, a sample of this interface for defining NEPs is shown. We can observe in it the graphical representations of the classes that compose the system.

It seems obvious in Fig. 2 that the complexity of writing the XML is quite higher than building the model just using the mouse with AToM^3. This difference is even

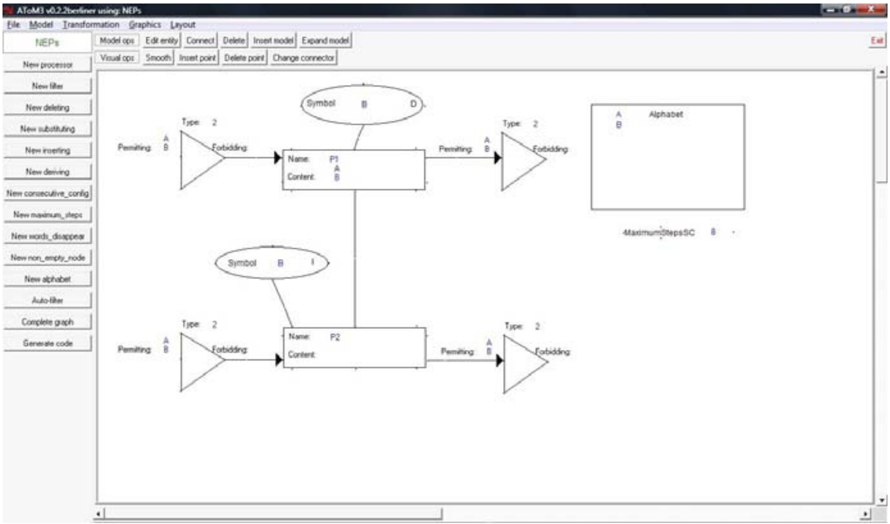


Fig. 2 The visual language in action

bigger when the end user is not used to write programs, and this is one of the system's biggest advantages together with the ability the user gets on seeing the model just at one sight (there is no need to inspect the code).

3 Conclusions and Further Research Lines

With this work we have completed a graphical wrapping to jNEP. The complete software platform includes the following modules:

1. jNEP, a Java NEPs simulator.
2. jNEPView, a Java graphical viewer of the simulations run by jNEP.
3. An AToM^3 domain specific visual language for designing NEPs.

All these modules have been developed by the authors of this paper. This work is focused on the last one.

Our final goal is to provide the users with an efficient platform to run NEPs and some graphical tools to ease the designing and debugging process. In the future we plan to refine these modules to add more features. Further on, we plan to develop a jNEP version that can be efficiently run by clusters of computers. jNEP currently includes a multithreaded version, but we want to explore other programming languages and concurrence primitives able to take more advantage of the cluster capabilities.

Acknowledgements. This work was partially supported by DGUI CAM/UAM, project CCG08-UAM/TIC-4425, by the R&D program of the Community of Madrid (S2009/TIC-1650, project "e-Madrid"); as well as by the Spanish Ministry of Science and Innovation, project "METEORIC" (TIN2008-02081), and sabbatical grant PR2009-0019.

References

1. Kelly, S., Tolvanen, J.-P.: Domain-Specific Modeling: Enabling Full Code Generation. Wiley-IEEE Computer Society (2008)
2. de Lara, J., Vangheluwe, H.: AToM^3: A tool for multi-formalism and meta-modelling. In: Kutsche, R.-D., Weber, H. (eds.) FASE 2002. LNCS, vol. 2306, pp. 174–188. Springer, Heidelberg (2002)
3. Ehrig, H., Ehrig, K., Prange, U., Taentzer, G.: Fundamentals of algebraic graph transformation. Springer, Heidelberg (2006)
4. Csuhaj-Varju, E., Salomaa, A.: Networks of parallel language processors. In: Păun, G., Salomaa, A. (eds.) New Trends in Formal Languages. LNCS, vol. 1218, pp. 299–318. Springer, Heidelberg (1997)
5. Csuhaj-Varjú, E., Mitrana, V.: Evolutionary systems: A language generating device inspired by evolving communities of cells. Acta Informatica 36, 913–926 (2000)
6. Castellanos, J., Martín-Vide, C., Mitrana, V., Sempere, J.M.: Solving NP-complete problems with networks of evolutionary processors. In: Mira, J., Prieto, A.G. (eds.) IWANN 2001, Part I. LNCS, vol. 2084, p. 621. Springer, Heidelberg (2001)
7. Hillis, W.D.: The Connection Machine. MIT Press, Cambridge (1985)
8. Errico, L., Jesshope, C.: Towards a new architecture for symbolic processing. In: Plander, I. (ed.) Artificial Intelligence and Information-Control Systems of Robots 1994. World Sci. Publ., Singapore (1994)
9. Paun, G., Rozenberg, G., Salomaa, A.: DNA Computing. New Computing Paradigms. Springer, Berlin (1998)
10. Paun, G.: Computing with membranes. Journal of Computer and System Sciences 61, 108–143 (2000)
11. Csuhaj-Varjú, E., Dassow, J., Kelemen, J., Paun, G.: Grammar Systems. Gordon and Breach, London (1993)
12. Castellanos, J., Martin-Vide, C., Mitrana, V., Sempere, J.M.: Networks of evolutionary processors. Acta Informatica 39(6-7), 517–529 (2003)
13. del Rosal, E., Nuez, R., Castaeda, C., Ortega, A.: Simulating neps in a cluster with jnep. In: Proceedings of International Conference on Computers, Communications and Control, ICCCC 2008 (2008)
14. Cuellar, M., del Rosal, E.: jNEPView: A graphical trace viewer for the simulations of nEPs. In: Mira, J., Ferrández, J.M., Álvarez, J.R., de la Paz, F., Toledo, F.J. (eds.) IWINAC 2009. LNCS, vol. 5601, pp. 356–365. Springer, Heidelberg (2009)

An Evolutionary Confidence Measure for Spotting Words in Speech Recognition

Alejandro Echeverría, Javier Tejedor, and Dong Wang

Abstract. Confidence measures play a very important role in keyword spotting systems. Traditional confidence measures are based on the score computed when the audio is decoded. Classification-based techniques by means of Multi-layer Perceptrons (MLPs) and Support Vector Machines have shown to be powerful ways to improve the final performance in terms of hits and false alarms. In this work we evaluate a keyword spotting system performance by incorporating an evolutionary algorithm as confidence measure and compare its performance with traditional classification techniques based on MLP. We show that this evolutionary algorithm gets better performance than the MLP when False Alarm (FA) is high and always performs better than the confidence measure based on the single score computed during the audio decoding.

1 Introduction

Speech recognition converts spoken words to text. Generally, the automatic speech recognition (ASR) systems try to identify all the words of a target language and produce an output consisting of the words found in the speech signal along with the initial and end times of each. Keyword spotting, as a task within speech recognition, deals with the search of a predefined set of terms within the speech signal. Therefore, it differs from the ASR systems due to just a few terms are important to be identified from the speech signal. Traditionally, keyword spotting systems use a dictionary

Alejandro Echeverría Rey
Universidad Autónoma de Madrid, Spain
e-mail: alejandro.echeverria@uam.es

Javier Tejedor
Universidad Autónoma de Madrid, Spain
e-mail: javier.tejedor@uam.es

Dong Wang
University of Edinburgh, UK
e-mail: dwang2@inf.ed.ac.uk

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 419–427.
springerlink.com

composed of these important words, modeled from their phone transcriptions, plus filler models to deal with the non-relevant words of the speech signal [6, 12, 11]. Therefore, their output is composed of the putative occurrences got from the decoding process plus filler models, which are automatically rejected. The main drawback of these systems is the need of re-processing the audio as soon as the list of terms changes, which is the most time-consuming task in a keyword spotting system and impractical when processing hundreds of hours. Alternatively, other keyword spotting systems index the audio in terms of sub-word units (commonly in the way of a lattice, i.e, a graph composed of nodes and arcs that connect the nodes) and next integrate an algorithm to search for the list of terms from such lattice. Although these systems have poorer performance than the filler model-based approaches due to they do not make use of any lexical information, there is no need of re-processing the audio when the list of terms changes, which is, in practice, highly valuable.

In keyword spotting, there are two different classes from which the performance can be measured. An occurrence detected by the keyword spotting system is a ”hit” if it appears in the speech signal between the time interval given and a ”False Alarm (FA)” when does not. It is straightforward to conclude that a keyword spotting system should produce as many hits as possible while minimizing the number of FAs. Confidence measures based on this idea haven been proposed to improve the system performance. Some of them use the confidence (score) computed during the decoding process of that the occurrence is actually a hit [11, 4], rejecting those whose score falls below a predefined threshold. On the other hand, as the occurrences belong to one of these two classes, hit or FA, confidence measures based on classification techniques have been also studied (e.g. Neural Networks (NNs) and Support Vector Machines (SVMs) [8, 1]) to reject those classified as FA.

On the other hand, evolutionary computation is a global optimization technique broadly applied to diverse fields [5] like schedule optimization, robot navigation, controller design, image processing, discrimination and classification, and so on. Therefore, as the final decision of accepting or rejecting a putative occurrence in keyword spotting is a classification problem, we propose the use of an evolutionary algorithm called Evolutionary Discriminant Analysis (EDA) [10] to classify the occurrences found during the search in lattice with the purpose of rejecting those classified as FA. We compare it with traditional techniques based on NNs (MLP) and on the score computed during the decoding process.

This paper is divided as follows: After this introductory section, the Speech recognition task is introduced in section 2. Section 3 explains how the confidence measures are built. In section 4, the experiments are presented. The last section of the paper presents the conclusions and opens some areas for future work.

2 Speech Recognition

We have focused on the keyword spotting task related to speech recognition. The keyword spotting system consists of three different steps:

Lattice generation: The audio files are decoded in terms of sub-word units (phones in our case) and the phone lattice is stored. Figure 1 shows an example of a phone lattice. The HTK tool [14] has been used for the lattice generation, where the Viterbi algorithm decodes the speech signal and generates the lattice. The lattice is generated by running the Viterbi algorithm in N-best mode and a depth of N=5 was found suitable in preliminary experiments.

Lattice search: An algorithm that searches for the actual phone transcription of each term in the lattice is run in order to generate the putative detections. Along with the detection of each term, and the initial and end times got from the initial and end times of the first and last phone of the term detected, this algorithm also computes a score. It represents the confidence of that each detection is a hit, using the language model score and acoustic score of each phone of the detection. These two values were stored in the lattice during the lattice generation. Readers are referred to [11] for more information about the score computation. The *Lattice2Multigram* implementation provided by the Speech Processing Group, FIT, Brno University of Technology has been used in this step.

Confidence measure: This module classifies the occurrences detected by the previous step in hit or FA and rejects those classified as FA to present the final list of occurrences.

Figure 2 shows the different modules of the whole system.

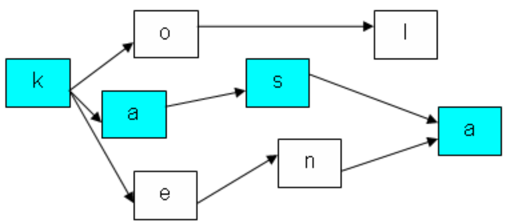


Fig. 1 An example of a lattice where the term “casa”, whose phone transcription is “k a s a”, has been found by the *lattice search* step

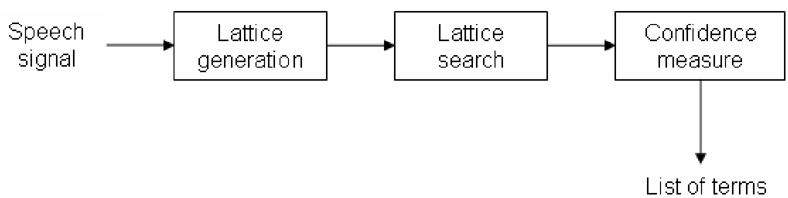


Fig. 2 Modules of the keyword spotting system, which receives a speech signal and outputs a list of terms found in it

3 Confidence Measure

3.1 Feature Selection

In building each confidence measure, three term-dependent features have been chosen as an input super-vector to the classification algorithm. It has been proved their good performance in hit/FA classification [13]. These features are:

Score: It represents the confidence of each detection computed as in [11].

Effective occurrence rate: It is defined as illustrated in Equation 1.

$$R_0(K) = \frac{\sum_i c_f(d_i^K)}{T} \tag{1}$$

Effective FA rate: It is illustrated in Equation 2.

$$R_1(K) = \frac{\sum_i (1 - c_f(d_i^K))}{T} \tag{2}$$

where T is the length of the audio processed, K is the term detected and $c_f(d_i^K)$ is the feature score for the detection i of the term K in equations 1 and 2.

The feature distribution represented in Fig.3 shows a very large overlapping between the two different classes (hit and FA).

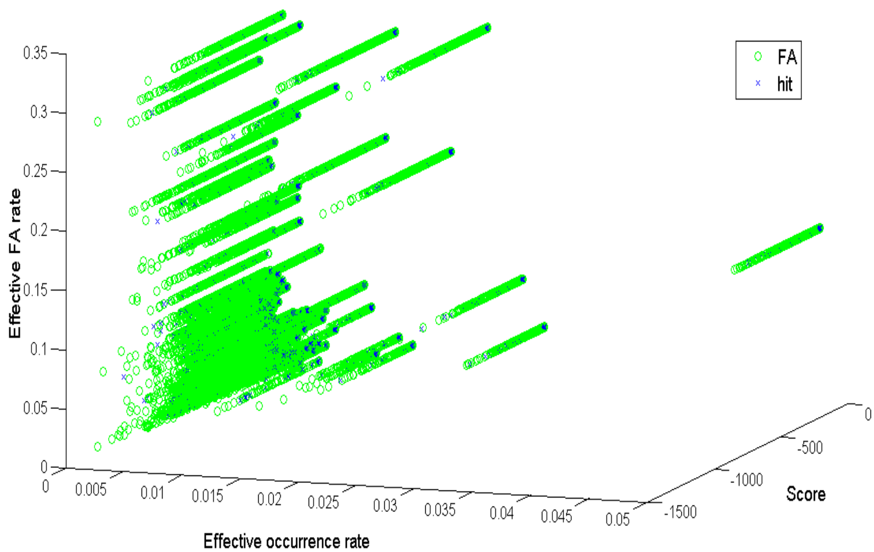


Fig. 3 Three dimensional graphic representation of the test set detections. For each detection (false alarm or hit), the three axis represent the three term-dependent features. Circles represent false alarms and crosses represent hits.

3.2 MLP-Based Classification

A 3-layer MLP was trained to build the MLP-based confidence measure from the features explained above. The structure of the MLP is composed of an input layer, a hidden layer and an output layer. The input layer has three units (in accordance with the number of features). The hidden layer has 5 hidden units and uses a sigmoid activation. The output layer has two units (hit and FA) and uses a softmax activation to achieve a reliable probability, which makes possible the final hit/FA classification. It must be noted that by using the softmax function, we force the sum of the output units to 1, thereby making the MLP a probabilistic classifier. The standard error backpropagation algorithm was used to train the model, as described in [3].

3.3 Evolutionary Algorithm-Based Classification

In our evolutionary confidence measure we use EDA to minimize the number of classification errors and to classify a set of detections as hit or FA using its output projection. Since the number of errors is a discrete non-differentiable quantity, it has to resort to an optimization method such as an evolution strategy [2] to minimize it. The algorithm consists of the following steps:

- An initial random projection is generated.
- The projection's worth (fitness) is the number of errors committed by assigning training patterns to the class of the closest mean in the projected space.
- The following steps are repeated until a prescribed number of generations without improvements is reached:
  - A new projection is generated by adding independent normal perturbations to each component of the current projection.
  - The number of misclassified patterns is calculated for the new projection.
  - The new projection becomes the current one when the error is reduced.
- The current projection is output, and the data will be classified as belonging to the closest projected train mean.

4 Experiments

4.1 Experimental Setup

The input acoustic signal is sampled at 16kHz and stored with 16 bit precision. Mel Frequency Cepstral Coefficients (MFCCs) were computed at 10ms intervals within 25ms Hamming windows. Energy and first and second order derivatives were appended giving the 39 MFCCs used to represent the signal.

The set of 47 phones in Spanish language [9] has been used during the lattice generation. Hidden Markov Models (HMMs) were used as acoustic models to represent the set of phones and they were context-dependent with 8-components Gaussian Mixture Models (GMMs).

The Spanish Albayzin database [7], which contains two different sub-corpora, has been used in the experiments: a phonetic corpus and a geographic corpus. Each contains a training set and a test set. The training of the HMMs was made from the ***phonetic training set***. A bigram was used as language model during the phone recognition process (lattice generation). It was built from the ***phonetic training set*** as well. The number of components GMMs in the context-dependent acoustic models was tuned for phone accuracy on the ***phonetic test set***. The parameters *word insertion penalty* and *language scale factor*, used within the decoding process, were tuned on the ***geographic training set***. Finally, the ***geographic test set*** was used as test set for the system evaluation.

To train the MLP and EDA we have used the ***geographic training set*** and a list composed of 500 terms, which appeared 12651 times in this corpus to ensure an enough set of examplers for training. To solve the imbalance between classes (hits and FAs), we have duplicated the examplers of the minority class (hit) to equal the number of hits and FAs used to train both the MLP and EDA. The parameter tuning for the MLP used the same ***geographic training set*** but a different list of terms, composed of 105 terms, appearing 10423 times to ensure a reliable parameter estimation. In tuning the parameters for the MLP, the number of hidden units (5 in our case) was chosen to maximise the classification accuracy by cross-validation in the ***geographic training set***. We have used the original EDA parameters: an $(15,100|2)$-ES with mutation step $\sigma = 0.15$ and 100 generations without improvements. Therefore, it was not necessary to make any tuning in this algorithm.

To test the system we have selected 400 terms appearing 11331 times in the ***geographic test set***. The list of terms used for training and parameter tuning had no overlapping between them. In order to simulate a real environment, the test set (400 terms in total) contained only 229 terms common to the training set.

4.2 Results and Discussion

Results are presented using the miss ratio (%miss) and FA ratio (%FA) widely used for speech recognition and keyword spotting tasks. These values are defined as follows:

$$miss\ ratio = 1 - hits\ ratio \tag{3}$$

where *hits ratio* is defined as follows:

$$hits\ ratio = \frac{Number\ of\ occurrences\ correctly\ detected}{Number\ of\ actual\ occurrences\ in\ the\ audio} \tag{4}$$

$$FA\ ratio = \frac{Number\ of\ occurrences\ detected\ that\ do\ not\ appear\ in\ the\ audio}{Number\ of\ non\ relevant\ words\ in\ the\ audio} \tag{5}$$

Figure 4 represents the curves plotting these two values, in which the performance of each confidence measure for different operating points is presented. The curves are got by varying the threshold to remove those terms found by the lattice search algorithm whose score remains below it. Therefore, the standard output of the lattice search step can be considered as the confidence measure based on the score in this representation, and the MLP and EDA confidence measures also incorporate it to make possible the whole curve computation.

The three curves show the improvement got by the MLP and EDA-based classification techniques over the score-based confidence measure alone, as they produce the same number of FAs and more hits than the latter for every operating point when the threshold is used to compute the whole curve. It means that actually both MLP and EDA classify some detections as FA that the score-based confidence measure alone is not able to reject. It should be also considered that the end of the MLP and EDA curves represents the number of hits and FAs that remain in the system before applying any threshold (i.e., the direct output after the confidence measure) and at this point, both the MLP and EDA outperform the rate achieved by the score-based confidence measure. It is also shown the improvement achieved by the EDA confidence measure over the MLP-based classification when the FA is high and the contrary when the FA is low. It means that the MLP is able to reject FAs with a high score, and EDA keeps more hits with a low score. It should be considered that the EDA implementation makes use of a linear implementation in classifying the detections as hit or FA, and in Figure 3, it is seen that both the hits and FA have a very large overlapped margin. It may explain the worse performance of the EDA than the MLP when the FA is low.

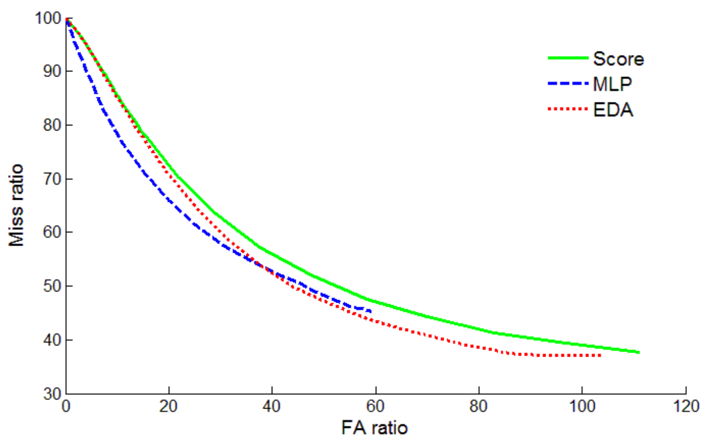


Fig. 4 Results in terms of *FA ratio* and *Miss ratio* for each confidence measure (Score, MLP and EDA) in the test set

5 Conclusions

Confidence measures play a very important role in keyword spotting and those based on classification techniques have shown to be an important contribution to increase the system performance. We have presented an evolutionary algorithm integrated into the framework of the confidence measures for a keyword spotting system to reject those keywords that are classified as FA. We have compared its performance with an MLP and a score-based confidence measure and we have shown that the EDA outperforms the rate achieved by the score-based confidence measure for every operating point and performs better than the MLP when the FA is high.

Our evolutionary procedure can be improved in several ways. First of all, we can make use of the non-linear version of EDA. Besides, designing the fitness function to maximize the number of hits may keep more putative detections (therefore more hits) which, along with other new confidence measures, may lead to a better improvement in the keyword spotting system. We are currently working on this point and hope to report the results in the near future.

Acknowledgements. The authors acknowledge support from the Spanish Ministerio de Ciencia e innovacin, project TIN2007-66862-C02-02, and from CAM/UAM, project number CCG08-UAM/TIC-4428.

References

1. Ben Ayed, Y., Fohr, D., Haton, J.P., Chollet, G.: Keyword spotting using support vector machines. In: Sojka, P., Kopeček, I., Pala, K. (eds.) TSD 2002. LNCS (LNAI), vol. 2448, pp. 285–292. Springer, Heidelberg (2002)
2. Beyer, H.G., Schwefel, H.P.: Evolution Strategies. A Comprehensive Introduction. Natural Computing 1, 3–52 (2002)
3. Bishop, C.M.: Neural Networks for Pattern Recognition. Oxford University Press, Oxford (1995)
4. Cuayáhuitl, H., Serridge, B.: Out-of-vocabulary Word Modeling and rejection for Spanish Keyword Spotting Systems. In: Coello Coello, C.A., de Albornoz, Á., Sucar, L.E., Battistutti, O.C. (eds.) MICAI 2002. LNCS (LNAI), vol. 2313, pp. 156–165. Springer, Heidelberg (2002)
5. Goldberg, D.E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley, Reading (1989)
6. Kim, J.G., Jung, H.Y., Chung, H.Y.: A Keyword Spotting approach based on Pseudo N-gram language model. In: SPECOM, pp. 156–159 (2004)
7. Moreno, A., et al.: Albayzin Speech Database: Design of the Phonetic Corpus. In: Eurospeech, vol. 1, pp. 653–656 (1993)
8. Ou, J., Chen, C., Li, Z.: Hybrid neural-network/HMM approach for out-of-vocabulary words rejection in mandarin place name recognition. In: ICONIP (2001)
9. Quilis, A.: El comentario fonológico y fonético de textos. ARCO/LIBROS S.A. (1998)
10. Sierra, A., Echeverría, A.: Evolutionary Discriminant Analysis. IEEE Tran. Evo. Comp. 10(1), 81–92 (2006)
11. Szoke, I., et al.: Comparison of Keyword Spotting Approaches for Informal Continuous Speech. In: ICSLP, pp. 633–636 (2005)

12. Tejedor, J., Colás, J.: Spanish keyword spotting system based on filler models, pseudo N-gram language model and a confidence measure. IV Jornadas en Tecnología del Habla, 255–260 (2006)
13. Wang, D., et al.: Term-dependent confidence for Out-Of-Vocabulary Term Detection. In: Interspeech, pp. 2139–2142 (2009)
14. Young, S., et al.: The HTK Book V3.4. Microsoft Corp. and Cambridge Univ. Eng. Dep. (2002)

Intelligent Agents and Wireless Sensor Networks: A Healthcare Telemonitoring System

Ricardo S. Alonso, Oscar García, Carolina Zato, Oscar Gil,
and Fernando De la Prieta

Abstract. E-healthcare has acquired great importance in recent years and requires the development of innovative solutions. This paper presents a telemonitoring system aimed at enhancing remote healthcare for dependent people at their homes. The system deploys a Service-Oriented Architecture based platform over a heterogeneous Wireless Sensor Networks infrastructure. Furthermore, the information obtained by telemonitoring systems must be managed by intelligent and self-adaptable technologies to provide an adequate interaction between the users and their environment. In the proposed system, the WSNs platform is integrated with a multi-agent architecture so that information gathered by WSN nodes is managed by intelligent agents with reasoning mechanisms.

Keywords: Multi-Agent Systems, Service-Oriented Architectures, Wireless Sensor Networks, e-Healthcare.

1 Introduction

There is an ever growing need to supply constant care and support to the disabled and elderly, and the drive to find more effective ways of providing such care has become a major challenge for the scientific community. The World Health Organization has determined that in the year 2025 there will be 1 billion people in the world over the age of 60 and twice as many by 2050 [1]. Furthermore, over 20% of those people over 85 have a limited capacity for independent living,

Ricardo S. Alonso · Carolina Zato · Oscar Gil · Fernando De la Prieta
Departamento de Informática y Automática, Universidad de Salamanca,
Plaza de la Merced, s/n, 37008, Salamanca, Spain
e-mail: {ralorin,carol_zato,oscar.gil,fer}@usal.es

Oscar García
E.T.S.I. de Telecomunicación, Universidad de Valladolid,
Camino Viejo del Cementerio s/n, 47011, Valladolid, Spain
e-mail: oscgar@tel.uva.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 429–436.
springerlink.com

requiring continuous monitoring and daily assistance [2]. The importance of developing new and more reliable ways of providing care and support for the elderly is underscored by this trend, and the creation of secure, unobtrusive and adaptable environments for monitoring and optimizing healthcare will become vital.

Healthcare telemonitoring systems allow patient's state and vital signs to be supervised by specialized personnel from a remote medical center. Such systems usually consist of a home monitoring subsystem, a remote monitoring subsystem in the medical center and a network which interconnects both. A telemonitoring system for healthcare needs to continuously keep track of information about patients and their environment. The information may consist of many different parameters such as location, the building temperature or vital signs.

Most of the information can be collected by distributed sensors throughout the environment and even the patients themselves. To facilitate the deployment of these sensors, it is preferable to use Wireless Sensor Networks (WSNs) instead of wired networks [3]. In existing buildings, wiring is more uncomfortable and difficult than using wireless devices. In the case of the biomedical sensors, it could be quite bothersome for patients to wear a mesh of wires. One solution would be to use a Wireless Body Area Network (WBAN) formed by wearable sensors [4]. However, it is not enough to gather information about the patients and their environment, but that information must be processed by self-adaptable and dynamic mechanisms and methods that can react independently of each particular situation that arises. In this sense, agents and Multi-Agent Systems (MAS) [5] comprise one of the areas that contribute expanding the possibilities of these systems.

This paper presents a telemonitoring system aimed at enhancing e-healthcare for dependent people at their homes. The system utilizes the SYLPH (*Services laYers over Light PHysical devices*) [6] experimental architecture that integrates a SOA (Service-Oriented Architecture) approach with heterogeneous WSNs. The architecture provides the telemonitoring system with a flexible distribution of resources and facilitates the inclusion of new functionalities in highly dynamic environments. Unlike other SOA-WSNs architectures [7] [8], SYLPH allows both services and services directories to be embedded in nodes with limited computational resources regardless of the radio technology they use. Furthermore, SYLPH can be integrated with *Flexible User and ServIces Oriented multiageNt Architecture* (FUSION@) [9], an architecture that combines a SOA approach with intelligent agents for building highly dynamic systems. Thus, context-aware information gathered by SYLPH WSNs can be used by intelligent applications based on agents that use reasoning mechanisms [10] to adapt their behavior to the context using the collected information.

Next, the problem description is introduced and it is explained why there is a need for defining a new telemonitoring system. Later, Section 3 describes the telemonitoring multi-agent system, as well as the integration of the SYLPH platform and the FUSION@ architecture employed to implement the system. Finally, Section 4 depicts the conclusions and future work.

2 Problem Description

One of the key aspects for the construction of healthcare telemonitoring systems is obtaining information about the patients and their environment through sensors. Biomedical sensors (e.g. electrocardiogram, blood pressure, etc.) and automation sensors (e.g. temperature, light, etc.) differ significantly in how they collect data. On one hand, biomedical sensors obtain continuous information about vital signs that is important and should not be lost. On the other hand, automation sensors get information at a lower frequency than biomedical sensors [3] because it is generally less important than vital signs. In a telemonitoring scenario, it is necessary to interconnect WSNs from different technologies [4], so having a distributed platform for deploying applications over different networks facilitates the developers' work and the integration of heterogeneous devices.

There are several telemonitoring healthcare developments based on WSNs [4] [11]. However, they do not take into account their integration with other architectures and are difficult to adapt to new scenarios. This is because such approaches do not allow sensors and actuators to communicate directly with one another, and instead gather data in a centralized way. Excessive centralization of services negatively affects system functionalities, overcharging or limiting their capabilities. However, distributed architectures such as SOA [12] look for the interoperability amongst different systems, the distribution of resources and the independency of programming languages. Some developments try to reach integration between devices by implementing some kind of middleware, which can be implemented as reduced versions of virtual machines, middleware or multi-agent approaches [13]. However, these developments require devices whose microcontrollers have large memory and high computational power, thus increasing costs and size. These drawbacks are very important regarding healthcare scenarios, as it is essential to deploy applications with reduced resources and low infrastructural impact.

One of the most prevalent alternatives in distributed architectures is agents and Multi-Agent Systems. An agent can be defined as anything with the ability to perceive its environment through sensors and respond in the same environment through actuators, assuming that each agent may perceive its own actions and learn from the experience [5] [10]. The development of agents is an essential piece in the analysis of data from distributed sensors and gives them the ability to work together and analyze complex situations, thus achieving high levels of interaction with humans [14].

There are different technologies for implementing WSNs. The ZigBee standard allows operating in the ISM (Industrial, Scientific and Medical) band, which includes 2.4GHz almost all over the world. The underlying IEEE 802.15.4 standard is designed to work with low-power limited resources nodes. ZigBee adds network and application layers over IEEE 802.15.4 and allows more than 65,000 nodes to be connected in a mesh topology WSN. Another standard to deploy WSNs is Bluetooth. This standard also operates in the 2.4GHz band and allows creating star topology WSNs of up to 8 devices, one acting as master and the rest as slaves, but it is possible to create larger WSNs through devices that belong simultaneously to several WSNs. However, it is not easy to integrate devices from different technologies into a single WSN [15]. The lack of a common architecture may lead to additional costs due to the necessity of deploying interconnection elements amongst different WSNs.

The architecture for the telemonitoring system presented in this paper tackles some of these issues by enabling an extensive integration of heterogeneous WSNs and providing a greater simplicity of deployment, optimizing the reutilization of the available resources in such WSNs. The architecture integrates a SOA approach for facilitating the distribution and management of resources (i.e. services). Unfortunately, the difficulty in developing a distributed architecture is higher. It is also necessary to have a more complex system analysis and design, implying more time to reach the implementation stage. There are several developments to integrate WSNs and a SOA approach [8]. However, those developments do not consider the necessity of minimizing the overload of the services on the devices. In contrast, our solution allows the services to be directly embedded in the nodes and invoked from other nodes either in the same WSN or another WSN connected to the former. Furthermore, it specifically focuses on using devices with small resources to save energy, CPU time and memory size.

3 System Description

The proposed telemonitoring system makes use of SYLPH (*Service laYers over Light PHysical devices*) [6], an experimental architecture developed by the BISITE Research Group at the University of Salamanca, Spain. This architecture integrates a SOA approach over WSNs for building systems that combine devices from different network technologies, such as ZigBee or Bluetooth. A node in a WSN of a specific technology can then be connected to a node in another WSN of a different technology. In this case, both WSNs are interconnected through a set of intermediate gateways connected simultaneously to several wireless interfaces. SYLPH allows applications to work in a distributed way and does not depend on the lower stack layers related to the radio transmission (i.e. data link and physical layers) and the WSN formation (i.e. network layer). A SOA approach was chosen because such architectures are asynchronous and non-dependent on context (i.e. previous states of the system) [12]. Thus, devices do not continuously take up processing time and are free to perform other tasks or save energy.

SYLPH implements an organization based on a stack of layers [3]. Each layer in one node communicates with its peer in another node through an established protocol. The SYLPH layers are added over the application layer of each WSN stack, allowing SYLPH to be reutilized over different WSN technologies. SYLPH implements the *SYLPH Message Layer* (SML), the *SYLPH Application Layer* (SAL) and the *SYLPH Services Directory Sub-layer* (SSDS) [6]. The SML offers the upper layers the possibility of sending asynchronous messages between two nodes through the *SYLPH Services Protocol* (SSP). Such messages specify the origin and target nodes and the service invocation in a *SYLPH Services Definition Language* (SSDL) format. The SAL layers on two different nodes can directly communicate each other using SSDL invocations and responses which will be delivered encapsulated in SML messages through SSP. The SSDS is used by nodes for locating services on other nodes in the WSN. *SYLPH Directory Nodes* (SDNs) acts as directories of the services offered by other nodes. Thus, any node in the network can request a SDN for the location of a certain service by sending it a SSDL search request embedded in a SML message over SSP. As mentioned above, several heterogeneous WSNs can be interconnected using SYLPH. For this, special nodes called *SYLPH Gateways* are used. Thus, they can forward messages amongst the different WSNs to which they belong. From the SAL layer's point of view, there is no difference between invoking a service stored in a node in the same WSN and invoking another one stored in a node in a different WSN.

3.1 Integration of SYLPH and FUSION@

In order to interact with a SYLPH network from a system that is not made up of WSNs, it is proposed to use FUSION@ [14], a Multi-Agent architecture for distributed services and applications. FUSION@ (*Flexible User and ServIces Oriented multiageNt Architecture*) proposes a new perspective, where Multi-Agent Systems and SOA-based services are integrated to provide ubiquitous computation, ubiquitous communication and intelligent interfaces facilities. The FUSION@ framework defines four basic blocks: *Applications*, *Services*, *Agents Platform* and *Communication Protocol* [14]. Besides, there are seven pre-defined agents that provide the basic functionalities of the architecture: *CommApp*, *CommServ*, *Directory*, *Supervisor*, *Security*, *Admin* and *Interface*. *Interface Agents* were designed to be embedded in users' applications and are simple enough to allow them to be executed on mobile devices, such as cell phones or PDAs.

This way, we have designed two agents in SYLPH to interact with FUSION@: *SylphInterface* and *SylphMonitor*. The *SylphInterface Agent* allows the rest of agents to discover and invoke services offered by SYLPH nodes. Moreover, the *SylphInterface Agent* can offer services to the SYLPH nodes. To do this, the WSN node in the FUSION@–SYLPH gateway stores services entries on its SSDS table. As can be seen in [9], the *SylphInterface Agent* performs as a broker between SYLPH and FUSION@. The *SylphMonitor Agent* allows the agents platform to monitor the state and operation of the SYLPH network. Thus, *SylphMonitor Agent* monitors all the traffc (i.e. service invocations, responses, registrations or searches) in the SYLPH network. The node gathers all the invocations and forwards them to the *SylphMonitor Agent* running on the agents' platform. The same process is done for service responses, searches and registrations. The *SylphMonitor Agent* makes it possible to observe when a node is searching for a certain service in the network, the services offered by the nodes, and the contents of the SSDS entries tables stored in the SDNs.

3.2 Implementation of the Healthcare Telemonitoring System

The proposed telemonitoring multi-agent system makes use of several SYLPH WSNs for obtaining context information in an automatic and ubiquitous way. In

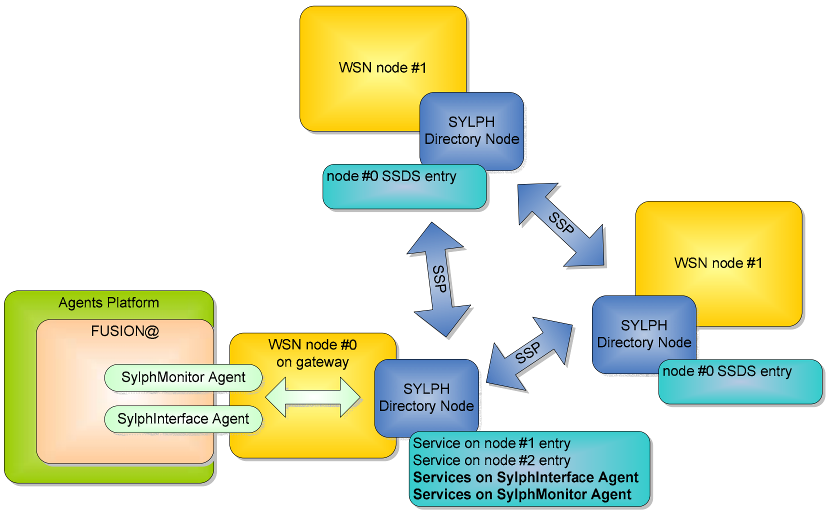


Fig. 1 Interaction between SYLPH and FUSION@

addition, the information gathered by the SYLPH nodes is managed by intelligent agents running in the system by means of the FUSION@ multi-agent architecture.

The system uses a network of ZigBee devices placed throughout the home of each patient to be monitored. Figure 2 shows the basic schema of the system. The patient carries a remote control (a small ZigBee device embedded in a wristband) that includes an alarm button which can be pressed in case of the need for remote assistance. There is a set of ZigBee sensors that obtain information about the environment (e.g. light, smoke, temperature, etc.) and react to changes (e.g. light dimmers and fire alarms). There are also several Bluetooth biomedical sensors placed over the patient's body. Each patient carries an Electrocardiogram (ECG) monitor, an air pressure sensor acting as respiration monitor, and a triaxial accelerometer for detecting falls. All ZigBee and Bluetooth devices can offer and invoke services within the network.

There is also a computer connected to a remote healthcare center via Internet for forwarding possible alerts to caregivers and allowing them to communicate with patients. This computer acts as a ZigBee coordinator and as master of the Bluetooth network formed by the biomedical sensors as slaves. At the SYLPH level, the computer works as a SYLPH Gateway so that it connects both WSNs each to other. In the other hand, there is a telemonitoring application based on the FUSION@ multi-agent architecture running on the computer in order to gather information from the SYLPH sensor nodes and send commands to the SYLPH actuator nodes. Such application is also the responsible for transfer the information between the SYLPH network and the other parts of the system.

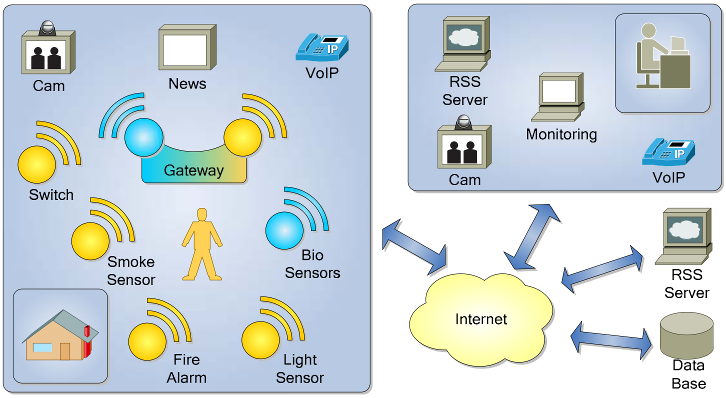


Fig. 2 Infrastructure of the telemonitoring system using SYLPH and FUSION@

4 Conclusions and Future Work

The integration of SYLPH and the FUSION@ multi-agent architecture allows developing highly-dynamic e-healthcare applications where context information gathered by heterogeneous WSNs is managed by intelligent agents. These intelligent agents can use reasoning mechanisms and methods in order to learn from past experiences and to adapt their behavior according the context. The use of a SOA-based approach provides a flexible distribution of resources and facilitates the inclusion of new functionalities in highly dynamic environments. Thus, functionalities are modeled as independent services offered by nodes in the network. These services can be invoked by any node in the SYLPH infrastructure, regardless the physical WSN which they belong. In addition, SYLPH nodes do not need large memory chips or fast microprocessors. The easy deployment of SYLPH-based systems reduces the costs in terms of development and infrastructure support.

Future work on SYLPH includes the development of nodes based on several WSN technologies (e.g. ZigBee and Bluetooth), as well as SYLPH Gateways for interconnect such kinds of WSNs. Other of the main aims is to achieve the integration of FUSION@ and SYLPH through the development of the *SylphInterface* and *SylphMonitor Agents* described above. We are also currently exploring alternative case studies for applying this platform and demonstrate that the approach presented is flexible enough to be implemented in other scenarios. However, one main issue to be taken into account is that the platform is still under development so it is necessary to define it through formal analysis and design methodologies and *Agent-Oriented Software Engineering* (AOSE) tools, such as Gaia or SysML.

Acknowledgments. This work has been supported by the Spanish Ministry of Science and Technology project TIN2006-14630-C03-03.

References

[1] World Health Organization, Global Age-Friendly Cities: A Guide (illustrated edn.). World Health Organization (2007)
[2] Erickson, P., Wilson, R., Shannon, I.: Years of healthy life. Healthy People 2000 Statistical Notes / National Center for Health Statistics (7), 1–15 (1995)
[3] Sarangapani, J.: Wireless Ad hoc and Sensor Networks: Protocols, Performance, and Control, 1st edn. CRC, Boca Raton (2007)
[4] Jurik, A.D., Weaver, A.C.: Remote Medical Monitoring. Computer 41(4), 96–99 (2008)
[5] Wooldridge, M.: An Introduction to MultiAgent Systems, 2nd edn. Wiley, Chichester (2009)
[6] Tapia, D., Alonso, R., De Paz, J., Corchado, J.: Introducing a Distributed Architecture for Heterogeneous Wireless Sensor Networks. In: Distributed Computing, Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assisted Living, pp. 116–123 (2009)
[7] Malatras, A., Asgari, A., Bauge, T.: Web Enabled Wireless Sensor Networks for Facilities Management. IEEE Systems Journal 2(4), 500–512 (2008)
[8] Marin-Perianu, M., Meratnia, N., Havinga, P., de Souza, L., Muller, J., Spiess, P., et al.: Decentralized enterprise systems: a multiplatform wireless sensor network approach. IEEE Wireless Communications 14(6), 57–66 (2007)
[9] Tapia, D., Bajo, J., Corchado, J.: Distributing Functionalities in a SOA-Based Multi-agent Architecture. In: 7th International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS 2009), pp. 20–29 (2009)
[10] Jennings, N.R.: Specification and implementation of a belief-desire-jointintention architecture for collaborative problem solving. Intl. Journal of Intelligent and Cooperative Information Systems 2(3), 289–318 (1993)
[11] Varshney, U.: Improving Wireless Health Monitoring Using Incentive-Based Router Cooperation. Computer 41(5), 56–62 (2008)
[12] Cerami, E.: Web Services Essentials: Distributed Applications with XML-RPC, SOAP, UDDI & WSDL, 1st edn. O'Reilly Media, Inc., Sebastopol (2002)
[13] Pavón, J., Gómez-Sanz, J., Fernández-Caballero, A., Valencia-Jiménez, J.J.: Development of intelligent multisensor surveillance systems with agents. Robot. Auton. Syst. 55(12), 892–903 (2007)
[14] Pecora, F., Cesta, A.: Dcop for smart homes: A case study. Computational Intelligence 23(4), 395–419 (2007)
[15] Cote, G., Lec, R., Pishko, M.: Emerging biomedical sensing technologies and their applications. IEEE Sensors Journal 3(3), 251–266 (2003)

Indoor Navigation Multi-agent System for the Elderly and People with Disabilities

Jose Maria Falco, Miguel Idiago, Armando Roy Delgado, Alvaro Marco, Angel Asensio, and Diego Cirujano

Abstract. The notion of Ambient intelligence (AmI) systems appeared with the main purpose to improve human-environment interaction by means of computerized pervasive environments. Rapidly the research community became aware of the potential of such systems to provide solutions in accordance with each individual, in particular elderly people and people with disabilities, which can undoubtedly benefit from many innovative services related to healthcare, location, security, etc.. In this paper we propose a indoor navigation system for smart buildings,for the elderly and people with disabilities which allows them to reach every place in the building following a personalized secure route. Guiding Under Individual Abilities (GUIA) has been designed taking into account elderly people and people with special needs on way-finding and moving in indoor environments.

Keywords: User-centred, guidance, navigation, service, agent, disabled, elderly.

1 Introduction

Social inclusion of people with special needs has therefore been a research issue for many years. In Europe, where there is more concern due to the ageing population, recent studies [1] point towards a social exclusion among the elderly and disabled people in both private and public environments (transportation, hospitals, etc). These risks are hard to avoid since they are specific to explicit people in particular situations, though these barriers can be addressed by means of applying AmI technologies.

Jose Maria Falco · Miguel Idiago · Armando Roy Delgado · Alvaro Marco · Angel Asensio · Diego Cirujano
Grupo Tecnodiscap, Universidad de Zaragoza, Maria de Luna 1
50018 Zaragoza, Spain
e-mail: {chefalco,midiago,aroy,amarco,aasensio, ciruman}@unizar.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 437–442.
springerlink.com

Large public buildings like museums or hospitals [2] are common targets for the deployment of AmI technologies; for example, a hospital can be rather unfriendly and can be like a labyrinth in the absence of some guidance support. A typical solution is to put signs, maps and guidelines where necessary in all public areas of the building; however elderly people or persons with disabilities may not be able to follow some of those signs. An alternative solution to approach everybody is to provide extra support with other people such as caretakers, but this approach may encroach on their independence and autonomy; besides allocating extra personal is too expensive.

To achieve a feasible solution it must be taken into account that each individual's unique combination of abilities and disabilities creates a “universe-of-one” situation[3]. Individual differences in user abilities need different guiding solutions: for blind users the information must be tactile and/or audio output; for wheelchair-bound users the way must provide ramps and elevators and minimal width of doors and corridors; for elderly and cognitive disabled guiding instructions should use common and uncommon landmarks and should be repeated with a certain frequency. The answer can be found in the development of guiding systems able to adapt to the special needs of each user. Our approach focuses in that matter and describes a solution based on an interoperability and collaboration schema of agents. All in unison will provide users with guidance to walk through the areas of effect safely.

As a summary, the paper is based on the description of the guidance system denominated GUIA [4]. It basically explains the agent specification of the system , interoperability schema and its differences in comparison with other navigation systems.

2 Appliying User-Centred Design to Guidance Systems

The development of indoor navigation systems is a flourishing research field, during the past few years several indoor navigation systems have been developed [5][6][8][9][10].. Typically indoor navigation is mainly focused on robot navigation like the navigation method developed for service robots by Jae-Han [6] et al.; however indoor navigation can perfectly be applied to wheelchairs [8][9] or for blind navigation [10].

Nevertheless only the system OntoNav [5] considers user abilities as one of the main factors to calculate the best route. Like OntoNav, GUIA is user centred at the time of calculating the appropriate route. The system provides with easy-to-understand instructions supported by building common landmarks - usually found in buildings - and any other special ones available - vending machines or coloured doors. We use the term messages to describe any information the system gives the user, either its output is text, sound, symbols, etc. The frequency of the guiding messages, contents, type, (icons, text, voice, photo...) and detail of the message also depend on the abilities of the user. Figure 1 shows the main features of GUIA:

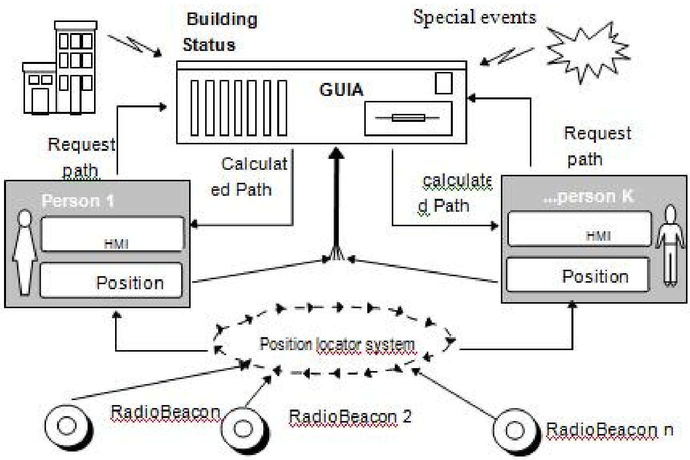


Fig. 1 GUIA system

The positioning system gives the (x,y,z) coordinates; such coordinates are translated by the locator to represent the exact point in the map of the building. Furthermore the coordinates of the user must also have an orientation that indicates his/her line of view. Altogether establishes a route between the different positions. After the route is planned, the system will transmit any necessary signals to the user automatically or by request while tries to reach the destination and it will also monitor if the user's position is consistent with the calculated route.

3 Implementation of the Multi-agent System

A multi-agent system require well-defined roles for each agent and a robust architecture where agent processes are related. Figure 2 shows the multi-agent architecture of GUIA:

According to the figure, system inputs are building areas where users want to go or to pass. Destiny information is joined with other information picked up from environment and message databases to bring dynamically the guiding message.

The involved agents in GUIA are:

- Trajectory Planner Agent - knows the condition of the building and user characteristics and calculates the optimal route.
- Path Builder Agent – draws the route map calculated according to the references and building signs.
- Orientator Agent - translates user's position to proper guiding messages to ensure the safety of the user as well as he can reach its destination and not to interfere in their personal autonomy.
- Tracker Agent - continuously verifies if the user is in the path calculated
- Locator Agent - detects the user's position and communicates it to the Tracker and the Orientator.

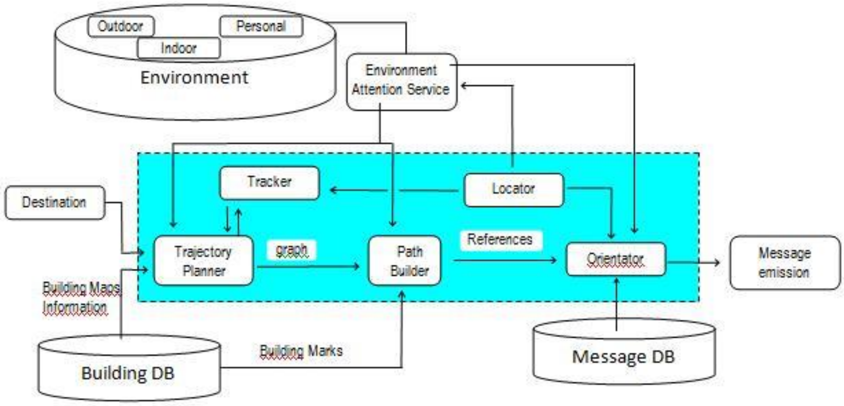


Fig. 2 GUIA agent architecture

As seen in figure 2 Locator Agent gets user position and orientation and communicates it to Orientator and Tracker agents. If the user is getting near the nest landmark or crosspoint Orientator sends a message; if not Tracker demands Trajectory Planner Agent a new route which is communicated to Path Builder. This one draws the new map and communicates it to the Orientator which sends a message to the user according to the new position.

The term Trajectory is the union of segments joining the starting position with the final destination including hallways, corridors, stairways, elevators, etc. Path is the translation of the trajectory in user understandable indications like “continues through this corridor", "when you pass the fire extinguisher, turn right", etc.

The first three agents work sequentially, in other words, the output of one module is the input of the next one; the last two ones, Locator and Tracker agents, operate in parallel with the others. Using information provided by the Locator, Tracker and Orientator agents check the position of each user with respect to the path indicated.

The tracker agent is capable of interrupting and restarting the guiding process. If it detects that the user is leaving the right path then informs the Orientator which emits an alert to the user and informs the Trajectory Planner.

The Orientator agent determines when you must provide guiding instructions. This agent indicates when to change direction or indicates relative goals using building indicators. It also indicates if the user has requested the proximity of public toilets, telephones, vending machines, information desks, or any other services in the building.

The parallel operation of these agents allows users to receive real time information about abnormal situations of the building (crowded locations, closed ways...) while the optimal route is recalculated from his current location.

GUIA agents are implemented using services based on the OSGi platform [7]. An agent in the system does not represent a service but the required software packages to perform the tasks. Services run underneath the agent abstraction level which, on the whole, helps to identify better the OSGi services independently. Services and agents are often related [11] in system developments but sometimes differ in the utilization according to the system needs. Our system can operate with any suitable accurate positioning service and is also being developed in tandem with the Location-Based Services of Marco et al.[12].

3.1 GUIA Routing Algorithm

GUIA is a hybrid navigation system and GUIA uses a hierarchical graph for representing the paths of all the floors in the building. If the floor where the user is (Ƒ0) is different from the floor where the user's target is (ƑT) GUIA uses the elevators/stairs hall in Ƒ0 as first target; then uses the elevators/stairs hall in ƑT as new source and the calculated best path Ƥ = Ƥ0 + ƤT.

GUIA uses an adaptation of the shortest path algorithm, which is a simpler algorithm than the graph traversal used by OntoNav. To get the optimal route GUIA's algorithm uses the user's target node as the Dijkstra's algorithm source node. The result is that GUIA knows the best path from every building graph node to the user's target node. If the user gets out of route GUIA locates him/her and offers the new route messages every time it is needed. In contrast OntoNav doesn't assume any action after the path calculation.

GUIA also takes into account user characteristics (cognitive level profile, physical abilities…) and building status (crowded corridors, elevator availability…) to calculate appropriate routes. The function which ponderates each edge of the building graph takes into account three parameters:

P(edgei)=f(architectural-element(i), user-abilities, legibility(i))

where **“legibility”** measures how many landmarks there are in **edgei**.

4 Conclusions

As a user centred approach. like OntoNav, the system has some useful features which makes a more friendly user-machine interaction like the optimization of the route and the customization of the messages according to user characteristics. Unlike OntoNav, GUIA also locates the user and recalculates the route dynamically. As a result can plainly be an indoor help navigation system or a comprehensive guiding system depending on user needs and preferences.

The main advantages of using Agents in the system are the increased dynamic interoperability between the elements as well as their higher autonomy. In addition the use of a OSGi bundle makes services suitable to any other bundles under the same OSGi framework, in the same way, GUIA can make use of other services like about location, mapping, lighting, person traffic, elevators, etc

GUIA needs an exhaustive evaluation with a wide sample of users in order to adjust the functions which calculate the route depending on the user's abilities, to accurate use guiding messages and to choose appropriate special landmarks or to eliminate inappropriate ones in order to improve its effectiveness. Some arrangements are being made with a residence nearby to start the testing.

Acknowledgments. This work was partially supported by the EU under MonAMI (IST-5-035147) and EasyLine+ (IST-045515) projects and by the Spanish Ministry of Science and Technology under the AmbienNET Project (TIN2006-15617-C03-02).

References

1. Jehoel-Gijsbers, G., Vrooman, C.: Social exclusion of the Elderly: a Comparative Study of EU Member States, ENEPRI Research Report, Centre for European Policy Studies, Brussels (2008)
2. Coronato, A., Esposito, M., Pietro, G.: A multimodal semantic location service for intelligent environments: an application for Smart Hospitals. Personal and Ubiquitous Computing 13(7), 527–538 (2009)
3. Kintsch, A., de Paula, R.: A framework for the adoption of assistive technology. In: SWAAAC 2002 (2002)
4. Falcó, J.M.: GUIA: Una aportación al guiado de personas mayores o con discapacidad, doctoral dissertation, University of Zaragoza (2004)
5. Anagnostopoulos, C., Tsetsos, V., Kikiras, P., Hadjiefthymiades, S.: OntoNav: A Semantic Indoor Navigation System. In: 1st Workshop on Semantics in Mobile Environments (SME 2005), Ayia Napa, Cyprus (2005)
6. Jae-Han, P., Seung-Ho, B., Moon-Hong, B.: An Intelligent Navigation Method for Services Robots in the Smart Environment. In: International Conference on Control, Automation and Systems, Seoul, pp. 494–497 (2007)
7. OSGi Platform: OSGi Initiative, OSGi Service Platform R3 (March 2004), http://www.osgi.org
8. Ding, D., Parmanto, B., Karimi, H.A., Roongpiboonsopit, D., Pramana, G., Conahan, T., Kasemsuppakorn, P.: Design Considerations for a Personalized Wheelchair Navigation System. In: 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, pp. 4790–4793 (2007)
9. Amaral, P.F.S., García, J.C., Bastos Filho, T.F., Mazo, M.: Ambient Assisted Route Planner Based on XML Files with Accessibility Information. In: 6th IEEE International Symposium on Intelligent Signal Processing Proceedings, Budapest, pp. 147–152 (2009)
10. Willis, S., Helal, S.: RFID Information Grid for Blind Navigation and Wayfinding. In: Proceedings of the Ninth IEEE International Symposium on Wearable Computers (ISWC 2005), pp. 34–37 (2005)
11. Bromuri, S., Urovi, V., Morge, M., Stathis, K., Toni, F.: A multi-agent system for service discovery, selection and negotiation. In: Proceedings of the 8th international Conference on Autonomous Agents and Multiagent Systems, Budapest, Hungary, May 10-15, vol. 2 (2009)
12. Marco, A., Casas, R., Falco, J., Gracia, H., Artigas, J.I., Roy, A.: Location-Based Services for Elderly and Disabled People. Computer Communications (2007)

Algorithms of Machine Learning for K-Clustering

S. José Luis Castillo, José R. Fernández del Castillo, and León González Sotos

Abstract. Machine learning is a scientific discipline that is concerned with the design and development of algorithms that allow computers to change behavior based on data, such as from sensor data or databases. They exist a number of authors have applied genetic algorithms (GA) to the problem of *K-clustering*, where the required number of clusters is known. Various algorithms are used to enable the GAs to cluster and to enhance their performance, but there is little or no comparison between the different algorithms. It is not clear which algorithms are best suited to the clustering problem, or how any adaptions will affect GA performance for differing data sets. In this article we shall compare a number of algorithms of GA appropriate for the *k-clustering* problem with some distributions of the collections Reuters 21578, including some used for more general grouping problem.

Keywords: Machine Learning, Data Mining, Evolutionary Algorithm, algorithms.

1 Introduction

Genetic algorithms have been successfully applied to problems in classification, modelling control [2], and in a considerable number of applications. In most cases, the key for success was the ability of GA to find exact or approximate solutions to optimization and search problems. GA is an iterative procedure which maintains a constant size population of feasible solutions, during each iteration step, called a generation, the fitness of current population are evaluated, and population are selected based on the fitness values. The motivation of using AG is that these they are a global and robust method of search of solutions, whose main advantage is the balance between the efficiency and effectiveness to solve different problems. Because, their search space should be defined inside a certain range, we analyze its behavior in problems of *k-clustering* and we analyze the adaptations for the *k-clustering* problem fall into the following areas: representation, fitness function, operators, and parameter values.

S. José Luis Castillo · José R. Fernández del Castillo · León González Sotos
Department of Computer Science
University of Alcalá
Campús Universitario s/n 28871 Alcalá de Henares, Madrid, Spain
e-mail: {jluis.castillo,joseraul.castillo,leon.gonzalez}@uah.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 443–452.
springerlink.com

2 Approaches of Genetic Algorithm

2.1 Representation

Genetic representations for clustering are based on two underlying schemes. The first allocates each object one (or more) integers or bits, known as genes, and uses the value of these genes to signify which cluster the object belongs to. The second scheme represents the objects with gene values, and the position of these genes signifies how the objects are divided amongst the clusters. Representations using these schemes differ in how the genes are assigned and how the gene values are interpreted. Figure 1 contains encodings of the clustering:{{ X1, X3, X6 },{X2, X4, X5}} for a number of representations that we will discuss in detail. The two clusters are denoted as 1 and 2 respectively, and the six objects (denoted by 1 to 6).

Group-number encoding [4] is based on the first encoding scheme and represents a clustering of n objects as a string of n integers where the ith integer signifies the group number of the ith object. When there are only two clusters this can be reduced to a binary encoding scheme by using 0 and 1 as the identifiers [12].

N x k matriz used by Bezdek et al [7], represent a clustering, which each row corresponding to a cluster and each column associated with an object. A 1 in row i, column j means that object j is in group i. Each column contains exactly one 1, whereas a row can have many 1's. All other elements are 0's. This representation can be adapted for overlapping clusters or fuzzy clustering [6].

Permutation with separators encodings uses the integers $n+1$ to $n+k-1$ to indicate where the cluster boundaries are in the permutation.

There are also permutation representations that need a local search to determine which clustering they correspond to – these are known as *greedy* representation - . *Greedy permutation* encoding [4] uses the first k objects in the permutation to seed k clusters. The remaining objects are then, in the order they appear in the permutation, added to the cluster which yields the best fitness value.

Bhuyan and Raghavan [9] also use a greedy encoding scheme. Here a permutation represents all possible clusterings with the correct number of clusters and the objects in that order. An algorithm by Fisher [13] is used to find which of these

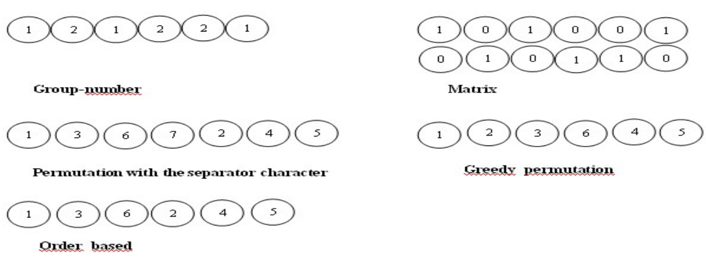


Fig. 1 Chromosomes representing the clustering $\{\{X_1, X_3, X_6\}, \{X_2, X_4, X_5\}\}$ for various encoding schemes

clusterings gives the best objective function value. This algorithm will find the optimal *k-clustering* for the permutation but the clustering is not necessarily unique. This is called *order-based* encoding. All of the above encoding schemes have some level of redundancy. We can swap the group numbers (or rows) k! ways, and the redundancy of permutation encoding grows exponentially with the number of objects [5].

2.2 Fitness Function

Objective functions used for traditional clustering algorithms can act as fitness function. However, if the optimal clustering corresponds to the minimal objective function value, we will need to transform the objective functions value since GA work to maximize their fitness values.

Krovi [12] uses the ratio of the between sum of squares and within sum of squares as his fitness function. Bhuyan et al [9] use the sum of squared Euclidean distance of each object from the centroid of its cluster. Alippi and Cucchiara [1] also use the same criterion, but use a GA that has been adapted to minimize fitness values. Bezdek et al's [7] clustering criterion is also based around minimizing the sum of squared distances of objects from their cluster centres, but they use three different distance metric (Euclidean, diagonal, and Mahalonobis) to allow for different cluster shapes.

2.3 Crossover

The crossover operator is designed to transfer genetic material from one generation to the next. The major concerns with this operator are validity and context insensitivity. *Group-number*: Single point and uniform crossover can be used for group-number chromosomes. Jones and Beltramo [4] compare three types of cross-over for the group-number representation: single-point, uniform, and edge-based. They use two versions of the former operators: one with rejection, and one with rejection and renumbering. Rejection is introduced to counter the problem of validity. A child is only accepted if it is valid.

Matrix: Alippi and Cucchiara [1] use a single point asexual crossover to avoid the problem of redundancy (figure 2). The tails of two rows of the matriz are swapped, starting from a randomly selected crossover point. Clustering with less than *k* groups may be produced by this operator.

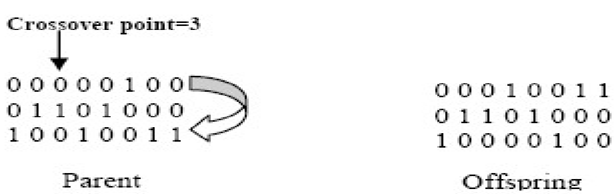


Fig. 2 Alippi and Cucchiara's asexual crossover

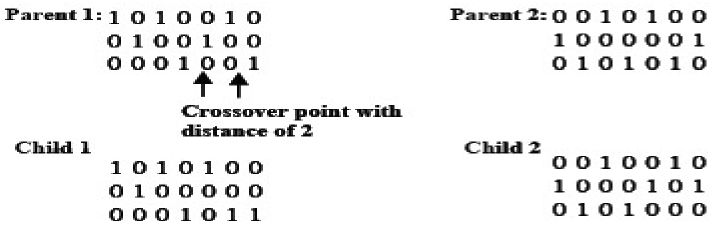


Fig. 3 Bezdek et al's 2-points matrix crossover

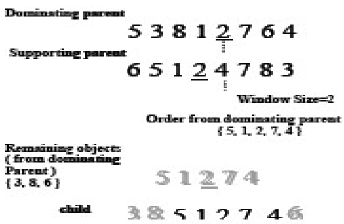


Fig. 4 Operator 1 Order-based

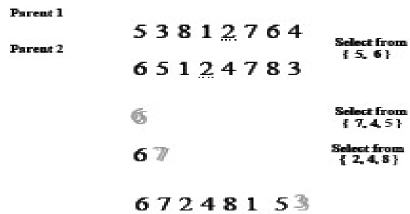


Fig. 5 Operator 2: Order-based

Bezdek et al [7] use a sexual 2-point crossover (figure 3). A crossover point and a distance (the number of columns to be swapped) are randomly selected.

Permutation with separators: Jones and Beltramo [4] use two crossovers for permutation encoding, PMX and OX [2][12]. In both cases parents are repeatedly crossed until the child decodes into a clustering with *k* groups.

Greedy permutation. Jones and Beltramo [4] also use PMX and OX crossover for the greedy permutation representation.

Order-based: Bhuyan et al [9] describe two operators for their order-based representation. Operator 1 (figure 4) randomly selects a dominant parent, the remaining parent becomes the supporting parent. Next an object and a distance or window size are selected at random. These define the substring which is copied from the supporting parent to the child. The substring is copied into the child so that the selected object is in the same position as in the dominating parent, and the order of the objects in the dominating parent is maintained. The remaining objects are placed into the child in the same order they appear in the dominating parent.

The Operator 2 (figure 5) starts by selecting the object in the first position of one of the parents, and placing this in the first position of the child. We then select an object from all the objects that are next to this object in either parent. The selected object becomes the second object in the child. This process continues until all of the objects are represented in the child.

Bhuyan's [8] croosover operator is similar to Operator 1 above, in that a substring from a supporting parent is copied to the dominating parent (figure 6).

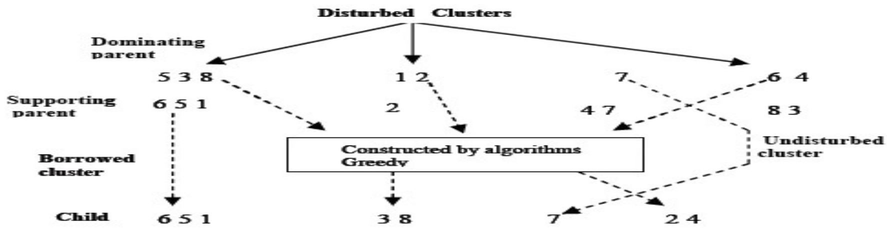


Fig. 6 Bhuyan's crossover

However, in this case we copy complete clusters (one or more) rather tan random substrings. First, the "borrowed" clusters are added to the dominating parent. Then we rearrange the remaining clusters in the dominating parent to make it a valid clustering.

2.4 Mutation

Mutation introduces new genetic material into the population. In a clustering context this corresponds to moving an object from one cluster to another. How this is done is dependent on the representation.

Group-number. Krovi [12] uses the mutation function implemented by Goldberg [2]. Here each bit of the chromosome is inverted with a probability equal to the mutation rate, P_{mut}. Jones and Beltramo [4] change each group number with probability, P_{mu}= 1/n where n is the number of objects.

Matrix. Alippi and Cucchiara [1] use a column mutation. An element is selected from the matrix at random and set to 1. All other elements in the column are set to 0. If th selected element is already 1 this operator has no effect.

Bezdek et al [7] also use a column matrix, but they choose an element that is currently 0 and set it to 1. The element that is 1 is set to 0.

Permutation with separators: Jones and Beltramo [4] randomly select two objects and swap them. To ensure that the resulting chromosome is valid, group separators cannot be swapped.

Greedy permutation: Two objects are randomly selected and swapped [4].

Order-based: The mutation operator used by Bhuyan, et al [9] is the same as that used for the greedy permutation representation and compares two mutation operators. The first moves a randomly selected object to a randomly selected cluster; the second moves the object only if the move results in a decrease in the fitness.

2.5 Parameter Values

The parameter values used for the genetic clustering implementations vary considerably, and there is little or no documented justification for the selection of these values. Some parameters values are even omitted. Population sizes range from 40 [8] [9] to 1000 [4], the number of generations varies from 40 [8] to 200 [7] or to complete convergence; crossover probabilities are: high (0.70, 1.0); and mutation rates are low (0.1, 0.2), although these values are high compared to typical mutation rates for genetic algorithms.

3 Experiments

3.1 Data Sets

In order to determine the best adaptions for the clustering problem, and the relationship between the data and these adaptions, we compared the performance of a range of adaptions over a number of data sets. We selected two data sets with differing numbers of objects and cluster. For each of these we generated a similar data set (same number of objects, attributes, and clusters; similar cluster shape and proximity). Exhaustive tests with different combinations of adaptions were performed on the generated data sets. Using the test result suitable adaptions were selected for the generated data sets. We then compared the performance of these adaptions. The selected data sets were Reuters, distribution 20 and 21 [11], the properties of data sets are listed in table 1. The Reuters data set contains a larger number of lexemas objects (also called root). Only 40 terms selected by the method lemmatisers nzipf [10].

Table 1. Data Sets

	Number of Objects	**Lexemas**	**Number of attributes**	**Number of Clusters**
Reuters 20	402	2908	40	26
Reuters 21	273	2527	40	16

The new data sets were generated with the same number of objects, attributes, and clusters as the original data sets. Further, the shape and proximity of the clusters were based on those found in the original sets. However, the range of attributes values, the number of objects in the individual clusters, and the relative positioning of the cluster was purposely altered for the generated data sets. Finally, the shape and positioning of the cluster was adjusted so that the correct clustering corresponded to the maximum value of the objective function.

3.2 Objective Function

The objective function for the experiment was trace (W) or the sum of the squared distances between objects and their cluster centres. This was minimized over the solution space. The attribute values were standardized to minimise the difference in objective function values.

3.3 Adaptions

The following adaptions were compared:

Representation: Both the group-number and order based representations were implemented. The poor performance of the GA with order-based representation meant that these trial took longer to complete, and due to time constraints the range of adaptions compared for this representation was reduced.

Fitness Function: Three different scaling mechanisms were used for the group-number representation: local transformation, adjusted transformation, lineal scaling with C_{mult}=2.0. Adjusted and lineal scaling were also compared for the order based representation.

Selection: Fitness proportional selection was compared with three different elite levels: the top 0,1 or 5% of population size individuals were copied straight from one generation to the next. Elite levels of 0 and 5 % were used with order-based.

Crossover: For the group-number representation, single point, uniform and edge-based crossover were compared. For order-based encoding the PMX crossover operator was compared with two new operators. The first, borrow, is similar to the operator described by Bhuyan [8], but the operator has been designed to avoid the need for local search. A single cluster is copied from the first parent and placed into the child. The remaining objects are added to the child in the order they occur in the second parent. The second operator is an edge-based operator for order-based encodings. This operator works exactly the same as for the group-number.

Mutation: The mutation implemented for the group-number representation randomly changes a group-number with probability P_{mut}/n. One type of mutation were compared for order-based encoding. *uniform*, moves an object with probability P_{mut}/n to a uniformly distributed point along the length of the chromosomes.

Parameters: The following parameters were compared for the group-number representation: population size {50,100,200,400}; crossover probability {0.50,0.70,0.90}; mutation rate {0.01,0.05,0.10,0.20}. For order-based these were reduced to: population size {50,100,200}, crossover probability {0.50,0.70,0.90}; and mutation rate {0.10,0.20}.

All of the GA used random initialization, and replaced the entire population during the reproduction phase (except in the case of elite selection). There were no checks to ensure that all chromosomes contained k groups. Since the operators are dependent on the representation type, the experimental tests were divided into two groups according to the representation type.

Each run of the AG continued until the correct clustering was found or the execution time exceeded five CPU minutes. The number of corrects runs and the average time to find the solution were recorded for each combination.

4 Results and Conclusions

The overall correctness results for the generated data (table 2 and 3) raise some interesting questions – why did the order-based AG fail to find a single solution for Reuters 20?. The large number of objects in the Reuters 20 data set may provide the answer.

A group-number representation for a data with a large number of cluster has a high level of redundancy. This means that the representation space has many more options than the solution space. This divides the focus of the population, and crossed between chromosomes near differing options tend to be unproductive and wasteful. Overall, redundancy slows down the convergence and reduces the diversity of the population.

Table 2 Corrections of group-number

		No of Corrects Runs					
		0	**1**	**2**	**3**	**4**	**5**
Reuters 20	**No of Trials**	456	70	61	64	157	2216
	% of Trials	15.1	2.3	2.0	2.1	5.2	73.3
Reuters 21	**No of Trials**	327	549	849	783	394	122
	% of Trials	10.8	18.2	28.1	25.9	13.0	73.2

Table 3 Corrections of order-based

		No of Corrects Runs					
		0	**1**	**2**	**3**	**4**	**5**
Reuters 20	**No of Trials**	1296	0	0	0	0	0
	% of Trials	100.	0.0	0.0	0.0	0.0	0.0
Reuters 21	**No of Trials**	578	56	40	48	66	508
	% of Trials	44.6	4.3	3.1	3.7	5.1	39.2

The poor performance of the order-based AG on Reuters 20 can be explained by the large number of objects in the data set, which affects the speed of the local search, and the size of the representation space (see table 4). This meant that the time limit of 300 seconds restricted the AG to only a few generations, limiting the AG's search to a very small portion of a very large representation space.

Table 4 Size of the representations space for generated data sets

	N	**K**	**Solution**	**Group-number Rep.**[1]	**Order-based Rep.**[2]
Reuters 20	150	3	$6x10^{70}$	$3x10^{71}$	$6x10^{266}$
Reuters 21	59	7	$1x10^{46}$	$7x10^{49}$	$2x10^{83}$

[1] Size of the group-number representation space is k^n. [2]Size of order-based representation space is ½ n(n+1)!

There appears to be advantages to using both clustering representations. Group-number encoding when combined with single point crossover is a fast method of finding clusterings for data sets with low *k*. This method also seems to be able to cope with relatively large data sets, although size of the data sets used in these experiments was limited. The performance of the group-number AG on data sets with high *k* values was poorer, with many AG returning sub-optimal minima. The performance of the order-based AG with edge-based crossover, as a result of the complexity of the local search, this method cannot cope with large data set, but is suitable for small data sets with large values of *k*.

GA should be modified in order to attain optimal performance for a particular data set. It may be possible to adapt GAs to suit a class of data sets (with approximately the same number of objects,clusters and distribution). Increasing the rate of mutation in the early generations (to increase the available genetic material during the initial stages), may improve performance in this case.

In conclusions the data sets require different adaptions for optimal performance (correctness and speed), this does not mean that a particular set of adaptions will not find the correct solution to a large number of data sets if given enough time. Indeed, it would seem that a number of adaptions (elite constant, fitness transformation, crossover probability, and mutation rate) have a large effect on the time taken to find the optimal solution, and that for simpler data sets these adaptions can be used to speed up the evolution of the GA without sacrificing correctness. Finally, the adaptions implemented in this experiment are only a sample of the numerous possible adaptions. Thus the performance of AG on these and other data sets may be improved by further adaptions.

References

1. Alippi, C., Cucchiara, R.: Cluster partitioning in image analysis classification. In: Proceedings Computer Systems and Software, pp. 139–144. IEEE, Los Alamitos (1992)
2. Golderg, D.E.: Genetic algorithm in search Optimization and Machine Learning. Addison Wesley Publishing Company, Inc., Reading (1989)
3. Olson, D., Delen, D.: Advanced Data Mining Techniques. Springer, Heidelberg (2008)
4. Jones, D., Beltramo, M.: Solving partitioning problems with genetic algorithms. In: Proceedings of the Fourth International Conference on GAs, pp. 442–449. Kauffman, San Francisco (1991)
5. Falkenauer, E.: The grouping GAs. Statistic and Computer Science, 79–102 (1993)
6. Bezdek, J.: Pattern recognition with Fuzzy Objetive algorithms, New York (1991)
7. Bezdek, J.: Genetic algorithm guided clustering. In: Proceeding First Conf. Evolutionary Computation IEEE World Congress on Computational Intelligence, pp. 34–39. IEEE, Los Alamitos (1994)
8. Bhuyan, J.: A combination of genetic algorithm and simulated evolution techniques for clustering. In: Proceeding of 1995 ACM Computer Science Conference, pp. 127–134 (1995)
9. Bhuyan, J., Raghavan, V.: Genetic algorithms for clustering with an ordered representation. In: Proceedings of the 4th International Conference on AG, pp. 408–415

10. Castillo, J.L., del Castillo, F., Gonzales, L.: Feature reduction for document clustering with NZIPF method. In: Proceedings of IADIS e-Society, vol. 25, pp. 205–209 (2009)
11. Lewis, D.D.: Reuters – 21578 text categorization test collection, http://www.daviddlewis.com/resources/testcollections/reuters21578 (cited July 2009)
12. Krovi, R.: Genetic algorithms for clustering. In: Proceedings of the 25th Hawaii Internat. Conference on System Sciences. IEEE Comp. Society Press, Los Alamitos (1991)
13. Fisher, W.D.: On grouping for maximum homogeneity. Journal of the American Statistical Association 53, 789–798 (1958); Cited in [9]

Monitoring and Adaptation of Assessment Process in Virtual Courses Based in Multi-agent Systems

M. Néstor D. Duque, C. Demetrio Ovalle, and B. Jovani Jiménez

Abstract. This paper presents an adaptive assessment system that adapts itself to the specific conditions of each student, modeled by means of a Multi-Agent Systems (MAS). This approach implies to see the monitoring and assessment as a teaching-learning tool and not only as an measuring mechanism. The Artificial Intelligence field, in particular the MAS approach plays a crucial role in the utilization of new technologies applied to Education and Learning Processes; the proposal advantage of the capacity to distribute knowledge and processing which is intrinsically provided by this Artificial Intelligence Technique. The literature reports important works most focused on the adaptation of the instruction processes, with very few supporting to learning assessment.

Keywords: Student monitoring, Multi-Agent systems, Adaptive Assessment, Virtual Education.

1 Introduction

From an educational point of view the curriculum planning comprises several issues as follows: the definition of general objectives, the establishment of the learning pedagogical activities, and the learning assessment. This last issue must be considered as a systematic and continuous process useful to validate the whole teaching/learning process. From a methodological perspective the assessment process can be seen as the result of comparing two situations: the student learning process observed and the expected [1]. Technological advancements and in particular those from Artificial Intelligence (AI) field have added flexibility and adaptability features within teaching/learning environments. Unfortunately, not every of profits earned by the instructional process have been transmitted at the assessment process, where tests have not already been personalized.

This paper aim at proposing and describing the main characteristics required by an assessment process which adapts itself to the academic and psycho-pedagogical

---

M. Néstor D. Duque · C. Demetrio Ovalle · B. Jovani Jiménez
Universidad Nacional de Colombia,
Sede Manizales. Sede Medellín, Colombia
e-mail: {ndduqueme,dovalle,jajimen1}@unal.edu.co

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 453–460.
springerlink.com

conditions, and preferences of the students within a framework of Virtual Courses. In addition, it exhibits an assessment module, part of an experimental platform of Adaptive Virtual Courses (SICAd) in research projects of GAIA (Research Group in Intelligent Adaptive Environments).

The rest of the document is organized as follows: section 2 outlines main elements of the assessment system proposal; section 3 describes the MAS proposed specifying system functionalities, section 4 presents the experimental platform, and the final section presents the conclusions and future work.

2 Adaptive Assessment in Virtual Courses

The adaptability of an educational system may be defined as the capacity of the system to dynamically adapt its behavior to the requirements of the student-system interaction. Adaptive and intelligent Web-based educational systems (AIWES) aim at providing students with an environment that reacts intelligently to the learners needs and incorporates their individual characteristics and situation by presenting appropriate suggestions, information, and teaching material in order to make learning more effective and easier for learners.

As it can be seen on figure 1, the components susceptible of adaptability in an education system may be the interfaces with the student, the lesson plan, the educational strategies, the filtering of information, and the evaluative process, which involve the integration of individual with collaborative learning activities.

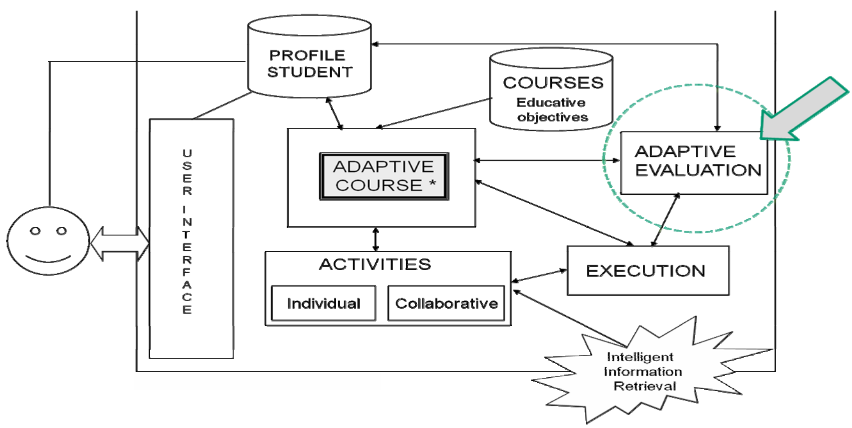


Fig. 1 Relationships among elements within an Adaptive Virtual Course System

The advancements in Artificial Intelligence Techniques and in particular their applications to adaptive educational systems provide good possibilities that researchers have already not completely exploited and presently become a field of special interest in order to implement solutions in automate way. A large numbers of works have been developed in instructional phase and the content presentation, but not every profits have been applied at the adaptive assessment process. In many cases, the assessment has been managed as a transposition of the traditional assessment and a few works have aimed to exploit the technological advantages to improve the evaluation process. The review of some related projects allows to see as the adaptive techniques for building knowledge assessment in virtual courses involves following tasks: Item Selection, Grading, and Student's Results Feedback [2],[3], [4], [5], [6], [7], [8], [9].

Our proposal incorporates and combines some of the elements found in the state of the art, but also extends the adaptation of the evaluation taking into account the different hierarchical levels of the Educational Objectives (EO) proposed, and psycho-pedagogic profile of the learner, in particular the learning style (LS). It is based on approach the monitoring and assessment as a teaching-learning tool and not only as a measuring mechanism.

According to Runyon and Von Holzen the online assessment needs to be viewed as an interactive mentoring opportunity that can be employed in online courses. The assessment techniques employed in an online course should be based on desired learning outcomes. Then these can be used by the students to evaluate their own progress through the course materials while also providing the instructor with evidence of the effectiveness of the course materials or indications of content areas that need further enhancement and/or development [10].

The assessment at different hierarchical levels of the educational objectives allows determine the specific problem and critical content may be reinforced. The LS is recognized as a hard differentiator in information processing by the apprentice. The LS may be taken into consideration for providing methods adequate of assessment to reflect student understanding of the learning outcomes. By monitoring the evolvement of the student, the system can redefine methods and strategies of assessment to meet the needs of students and customize the process.

In general form and based in the schema of ITS (Intelligent Tutoring System), the traditional models are complemented by a new model: Adaptive Assessment Model, as exhibits in Figure 2.

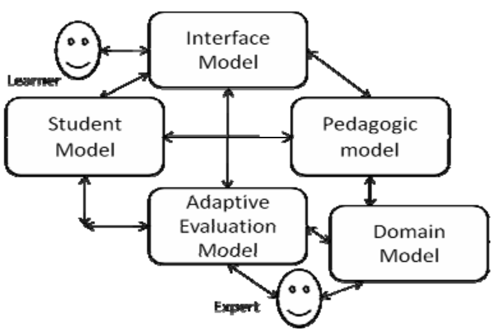


Fig. 2 Components involved in the proposed system

Proposing an Adaptive Assessment Model requires the precise definition of a variety of elements in such way that they may be handled in the adaptation process. On one hand, the adaptation task must define the relevant elements of the student profile that determine the customization, in this case are the academic trajectory (expressed in terms of EO) and LS (obtained by applying ILS and VARK test). On the other hand, the domain of the course and evaluation component must be represented in such way that it can be adapted according to the needs of the learners, specifying the components able to be adapted; in this case are EO to evaluate, proofs type, content of the proofs and specific strategy, according to the focus of the system.

In short, this proposal presents an adaptive assessment process which based into the preferences and academic conditions and psycho-pedagogical characteristics of the students. The assessment is oriented through EOs initially proposed into the course, which are expecting to be attained by the student within the teaching/learning process and recognizes the individual characteristics of learner.

According to the perspective given in [11], the course's domain follows a hierarchical structure and a fine level of detail is adopted in order to define the EO. The process must identify where the student presents failures. For doing so, the assessment is performed starting with the high level objective and then, the process continuous up-down reviewing sub-objectives till exactly locates the point where the student exhibits difficulties. This action allows re-planning the instruction for this specific learning problem case. Other issue that has to be considered in the assessment process is the level of difficulty of questions. The question banks for our system respects the norms established for this kind of repositories, i.e., metadata are included for each assessment question which is associated with EO to attain, learning styles supported, and the level of difficulty assigned, as a guide within the system assessment adaptation process.

3 MAS Proposal for Adaptive Assessment

The path to follow may be oriented by the disintegration into functional blocks, without loosing the systemic point of view, which leads to distributing the solution in diverse entities that require specific knowledge, processing and communication between each other. Having these characteristics, modeling the problem using a Multi-Agent System (MAS) seems to be a promising option. The main motivation for selecting MAS is the possibility to distribute the components of intelligence outlined by the solution on the proposed problem [12].

The MAS principles have shown an appropriate potential in the development of learning systems due to the fact that nature of the learning/teaching problems are more easily solved through a cooperative approach. Figure 3 exhibits the MAS´s architecture and the relationships among software agents.

Starting from the decision above, the design process for the system is initiated using the MASCommonKADS methodology as foundation, which defines the necessary models for the analysis and design phases and provides complete documentation; besides, other applications in similar cases report to have very positive

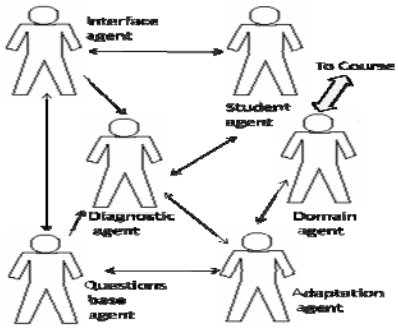


Fig. 3 Multi-Agent System in the proposal

results [13]. Nevertheless, the experienced acquired suggests including some models from other methodologies such as MaSE (Multi-agent system Software Engineering) [14] (Hierarchical Objective Diagrams as support in dividing some complex tasks) and GAIA [15] (Roles Model, important in clearly determining what can be expected from the agents).

The steps to follow in order to retrieve the personalized assessment to the student are based on the request of an external agent in order for the selection process of a certain student's assessment to be initialized (Adaptation Agent). After the student's profile data are collected (Student Agent), the system sends a question request to the Bank of Questions Agent, which is selected from the database of the bank of questions according to some criteria such as the educational objective to be evaluated, learning style of the student and preferences. After the question is presented to the student and this has given an answer (Interface Agent), the outcome is verified and the educational objective is fulfilled, or a re-adaptation process is initiated in order to detect faults through the whole process, which have not allowed the expected outcome.

The Student Agent allows handling diverse elements in the profile of the learner according to particular interests. Forms and tests supplied by the supporters of the included characteristics are used for input information, such as psychological tests, learning styles tests, sociograms, and some values directly obtained from the system during the interaction process. The Domain Agent manages the structure of the course associated to an acyclic graph whose nodes are the EO to be evaluated, while keeping information of the pedagogical resources.

The retrieval of the stored question is performed by the Bank of Questions Agent, which has only information about the way in which the resources will be received, permitting to use diverse visions for the resource composition, naming and allocation.

The Diagnostic Agent is an agent in charge of the assessment processes, performing tests, input tests, and so on; who contains the required knowledge to classify and locate the student.

These steps can be view in the sequence diagram in figure 4.

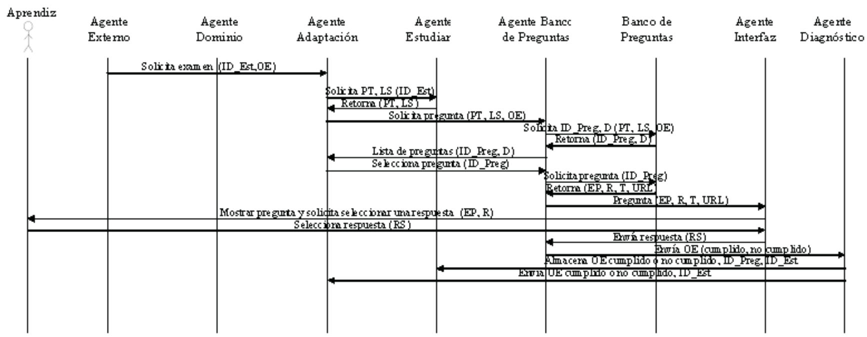


Fig. 4 Sequence Diagram in proposed model

4 SICAd Platform, Intelligent System of Adaptive Courses

Based on the previously established topics experimental platform Intelligent System of Adaptive Courses. SICAd, was constructed. It is completely functional and can be used for the assembly of diverse courses and applying different pedagogical trends. It was developed based on free and multiplatform tools; therefore it can be installed both on Windows and Linux environments and it allows access from any platform that has a Web browser.

The Web Server is Apache Software Foundation Tomcat, with support for the Database engine MySQL. The programs were developed in Java j2sdk1.6.0. The multi-agent platform was build on Java using the JADE framework [16] advantages that simplify the MAS construction and provides a set of tools for system debug. The integration between the presentation layer (JSP) and the Multi-Agent platform is done through the JadeGateway library offered by JADE.

The full MAS is presents in figure 5 highlighting the assessment module.

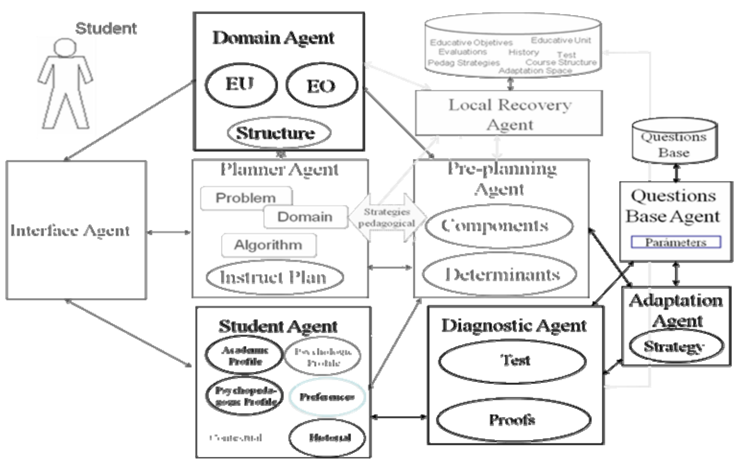


Fig. 5 Sequence Diagram in proposed model

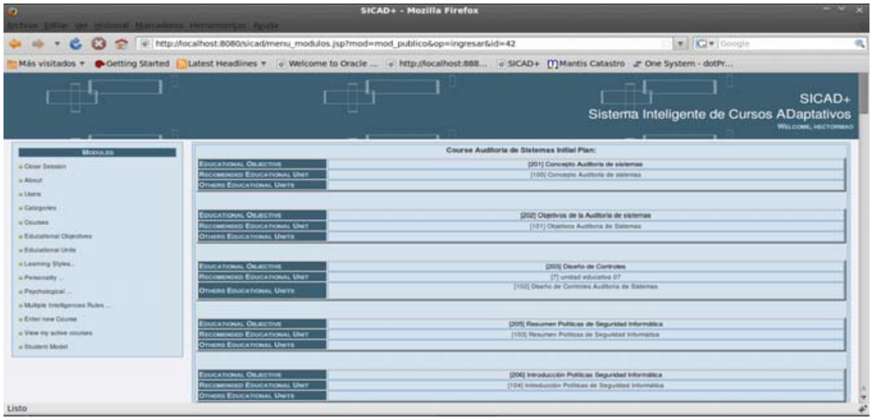


Fig. 6 Sequence Diagram in proposed model

The system permits to administrate elements of the virtual course: Users, Educational Objectives and Educational Units, assessment, to construct courses by means of the Educational Objectives that compose it and to automatically generate the domain of the planner and to execute it for the personalized generation of each course, as it is presented in figure 6, for case study.

5 Conclusions and Future Work

The proposal presented shows the possibilities that are offered when including in the courses an assessment scheme that emphasizes in the characteristic of readapting the plan according to the student, when the latter doesn't pass the threshold with the expected goals.

The requirement to distribute the necessary knowledge led to choose a MAS for modularizing and distributing the problem solution. The way that the problem is raised allows us to choose different choices in order to make the adaptation process which is presented as a neutral tool regarding pedagogical thoughts of those who implement it.

In this moment we are proving the interactions and evaluating the outcomes, looking towards the integration process with the experimental platform of virtual courses. In the proofs selection, the proposal has been validated obtaining the expected outcomes. Preliminary results are encouraging and demonstrate the important role of monitoring and evaluation in teaching-learning process, allowing the feedback and redefinition of strategies aimed at achieving the objectives.

It presents a difficulty in building a greater number of tests in accordance with the different levels of the educational objectives and the individual characteristics of the student.

References

1. Gagné, R.M.: Principios de aprendizaje para selección y uso de medios de instrucción. Editorial Diana. Octava Edición, México (1987)
2. Conejo, R., Millán, E., Cruz, J.L.P., Trella, M.: Modelado del alumno: un enfoque bayesiano. Revista Iberoamericana de Inteligencia Artificial 12 (2001)
3. Duque Méndez, N., Ovalle, D., Vicari, R., Silveira, R.: Modelo inteligente genérico para adaptatividad de cursos virtuales. Revista Novas Tecnologias na Educação, Brasil (2008)
4. Gonçalves, J.P.: A integraçao de Testes Adaptativos Informatizados e Ambientes Computacionais de Tareas para o aprendizado do Ingles instrumental. Trabajo de grado para optar para el grado Maestría en ciencias de Computación y Matemática Computacional (2004)
5. Papanikolaou, K., Grogoriadou, M., Kornilakis, H., Magoulas, G.D.: INSPIRE: An intelligent system for personalizad instruction in a remote environment. In: Third Workshop on Adaptative Hypertext and Hypermedia (2001)
6. Arroyo, I., Conejo, R., Guzman, E., Woolf, B.P.: An Adaptive Web-based Component for Cognitive Ability Estimation. In: Artificial Intelligent in Education: AI-ED in the Wired and Wireless Future. IOS Press, Amsterdam (2001)
7. Jiménez, M., Jiménez, J., Ovalle, D.: Online assessment for virtual adaptive intelligent tutoring courses using the multi-agent system model. In: TISE 2008, Santiago de Chile (2008)
8. Sosnovsky, S.: Adaptive Navigation for Self-assessment Quizzes. Springer, Heidelberg (2004)
9. Boticki, I., Budišcak, I., Hoic-Božic, N.: Module for online assessment in AHYCO learning management system. Novi. Sad. Journal of Mathematics 38 (2008)
10. Runyon, D., Von Holzen, R.: Assessment in Online Courses: Practical Examples. In: 18. Annual Conference on Distance Teaching & Learning Northwest Missouri State University (2005)
11. Méndez, D., Darío, N., Ramírez, J., Claudia, Guzmán, J.: IA Planning for Automatic Generation of Customized Virtual Courses. Frontiers in Artificial Intelligence and Applications, vol. 117. IOS Press, Amsterdam (2005)
12. Duque, Méndez, N., Ovalle, D., Silveira, R.: Adaptive virtual courses development based on Multi-agent system approach. In: IX World Conference on Computers in Education – WCCE (2009)
13. Iglesias, C.: Definición de una metodología para el desarrollo de sistemas multiagente. Tesis doctoral. Departamento de ingeniería de sistemas telemáticos. Universidad Politécnica de Madrid (1998)
14. Deloach, S.: Multiagent Systems Engineering of Organization-Based Multiagent Systems. In: Memorias 4th Intenacional Workshop on Software Engineering for Large-Scale Multi-Agents Systems Selmas 2005, St. Louis (2005)
15. Wooldridge, M., Jennings, N.R., Kinny, D.: The Gaia Methodology for Agent-Oriented Analysis and Design. International Journal of Autonomous Agents and Multi-Agent Systems (2000)
16. Bellifemine, G., Caire, C., Poggi, A., Rimassa, G.: JADE: Java Agent Development Framework. A White Paper. University of Parma (2003)

A Model-Based Ambient Agent Providing Support in Handling Desire and Temptation

Mark Hoogendoorn, Zulfiqar A. Memon, Jan Treur, and Muhammad Umair

Abstract. An ambient agent system is presented estimating a human's dynamics of desiring and being tempted. The agent is equipped with a dynamical model of the human's processes which describes how a desire relates to responses in the form of being prepared for certain actions, which in turn relate to feelings which can be biased, due to experiences in the past. It is shown how by use of this dynamical model, the ambient agent is able to predict and assess a human's desire state, and his or her preparation for certain actions, and use this assessment to suggest alternatives to avoid falling for certain temptations.

Keywords: Ambient Agent, Dynamical Model, Desire, Temptation.

1 Introduction

Agent modelling provides a useful design approach to the area of ambient systems [1,18]. One of the more ambitious challenges in this area is to create ambient systems with an appropriate human-awareness: awareness of the (mental) states of humans. Human-aware ambient systems can be taken to perform a certain type of mindreading or to possess what in the psychological and philosophical literature is called a Theory of Mind [12, 14]. During the evolutionary human history, mindreading has been developed to address different types of mental states, such as desire, intention, attention, belief or emotion states [12]. Inspired by such capabilities as developed in nature, ambient systems can be designed that have mindreading capabilities for one or some of these types of mental states. Such

Mark Hoogendoorn · Jan Treur
VU University, Department of Artificial Intelligence
De Boelelaan 1081, 1081 HV Amsterdam, The Netherlands

Zulfiqar A. Memon
Sukkur Institute of Business Administration (Sukkur IBA) Air Port Road Sukkur, Pakistan

Muhammad Umair
COMSATS Institute of IT, Department of Computer Science Lahore, Pakistan
e-mail: {mhoogen,zamemon,treur,mumair}@few.vu.nl

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 461–475.
springerlink.com

mental states can be dynamic and often interact with each other. To obtain an adequate human-aware ambient system, a dynamical model describing these dynamics and interaction is needed. To design an ambient system incorporating such a model, agent modelling offers a useful approach, as agents are able to integrate such dynamical models and reason about them [4]. Human-aware ambient agents equipped with the ability to reason about the different types of mental states can be applied to support of humans, for example persons vulnerable to temptations due to a developing addiction. More specifically, this paper focuses on the dynamics and interaction of an individual's desires and temptations and integrates a domain model for these dynamics in an ambient agent model to provide effective support by an enhanced awareness of the cognitive and affective states of the person. A desire triggers a number of responses in the form of preparations for certain actions related to the desire that result in certain feelings. In a reciprocal manner, the generated feelings affect the preparations; for some literature on such reciprocal interactions between cognitive and affective states, see, for example, [11, 17, 19].

In this paper, first in Section 2 the domain model for the dynamics of desires, preparations and feelings is described. Section 3 presents the ambient agent model which integrates the domain model. Section 4 presents some simulation results of the integrated ambient agent model. In Section 5 verification of the integrated model is addressed. Section 6 is a discussion.

2 Desires, Preparations and Feelings

Any mental state in a person induces emotions felt by this person, as described in [10]. Following [9, 10] it is assumed that responses in relation to a mental state of desiring roughly proceed according to the following causal chain: desire → preparation for response → body state modification → sensing body state → sensory representation of body state → induced feeling. An 'as-if body loop' uses a direct causal relation preparation for response → sensory representation of body state as a shortcut in the causal chain; cf. [9]. The body loop (or as-if body loop) is extended to a *recursive (as-if) body loop* by assuming that the preparation of the bodily response is also affected by the state of feeling the emotion: feeling → preparation for the bodily response. Such recursion is suggested in [10], noticing that what is felt is a body state which is under control of the person. Within the model used in this paper, both the bodily response and the feeling are assigned a level or gradation, expressed by a number. The activation of a specific action preparation is based on both the activation level of the desire and of the feeling associated to this action. This illustrates Damasio's theory on decision making with a central role for emotions felt, called the Somatic Marker Hypothesis [2,7, 8]. Based on the recursive *as-if body loop*, not only the strength of the connection from desire to preparation but also the strength of the connection from feeling to preparation will play an important role in deciding which action to pursue. When one or each of these connections is weak it will not lead to a high activation level of the preparation state, whereas a strong connection strength may result in a high activation level of the preparation state so that it may become the dominant option that can play the role of a strong temptation.

The strengths of the connections from feeling to preparation are subject to learning. Especially when a specific action is performed and it leads to a strong effect in feeling, by Hebbian learning [3, 15] this may give a positive effect on the strength of this connection and consequently on future activations of the preparation of this specific action. Through such a mechanism experiences in the past may have their effect on behavioural choices made in the future. The ambient agent uses the model and hence, it is expressed in the next section.

3 The Ambient Agent Model

Based upon the domain model as briefly expressed in the previous section, and ambient agent model has been developed. The ambient agent model was specified in LEADSTO [6], in which both logical and numerical relations can be specified. Moreover in this language, direct temporal dependencies between two state properties in successive states are modeled by *executable dynamic properties.* The LEADSTO format is defined as follows. Let α and β be state properties of the form 'conjunction of ground atoms or negations of ground atoms'. In the LEADSTO language the notation $\alpha \twoheadrightarrow_{e, f, g, h} \beta$, means:

> *If state property α holds for a certain time interval with duration g, then after some delay (between e and f) state property β will hold for a certain time interval of length h.*

Here, atomic state properties can have a qualitative, logical format, such as an expression desire(d), expressing that desire d occurs, or a quantitative, numerical format such as an expression has_value(x, v) which expresses that variable x has value v. The LEADSTO language features a simulation tool that is able to execute the dynamic LEADSTO properties.

Within the ambient agent model, the model for the dynamics of desires, preparations and feelings as expressed in Section 2 was embedded in order to enable the agent to reason about this process, and to assess the person's desires, preparations and feelings. In psychology, this capability is often referred to as mindreading or Theory of Mind [12]. The embedding uses the format that the causal relationships of the model described in Section 2 above are transformed into relationships for beliefs of the ambient agent on mental states of the person. In order to achieve this, the idea of recursive modelling is used; e.g., [16]. This means that the beliefs that agents have about each other are represented in a nested manner. To this end, each mental state is parameterized with the name of the agent that is considered, thus creating concepts like has_state(human, feeling(b, 0.5)) , has_state(AA, performed(suggest(X)). In addition, a number of meta-representations are introduced. For example, has_state(AA, belief(has_state(human, feeling(b, 0.7)))) states that the ambient agent (AA) believes that the human has a feeling level of *0.7* for b. The following are the resulting agent local properties (ALP) that specify the processes within the ambient agent. The step size is indicated by Δt. The first properties specify how the agent AA observes the human's body state and creates a belief about it.

ALP1 Observing the human's body state
If the human has certain body state, then the ambient agent AA will observe this.
has_state(human, body_state(B, V, t))
→ has_state(AA, observed(has_state(human, body_state(B, V, t))))

ALP2 Generating a belief for the human's body state
If the ambient agent AA observes that the human has certain body state,
then it will generate a belief on it.
has_state(AA, observed(has_state(human, body_state(B, V, t))))
→ has_state(AA, belief(has_state(human, body_state(B, V, t))))

The desire considered in the example scenario is assumed to be generated by sensing an unbalance in a body state *b*, according to the principle that organisms aim at maintaining homeostasis of their internal milieu. The first dynamic property addresses how body states are sensed. The following properties specify how the ambient agent observes and generates beliefs about the human's sensing and sensory representation process.

ALP3 Generating a belief for a human's sensing
If AA believes that the human has certain body state,
then it will generate a belief that after Δt the human will sense this body state
has_state(AA, belief(has_state(human, body_state(B, V, t))))
→ has_state(AA, belief(has_state(human, sensor_state(B, V, t+Δt))))

For the example scenario this dynamic property is used for *B* to estimate the person's sensing of the body state *b* from which the desire originates (e.g., a state of being hungry), and the body states b_i involved in feeling satisfaction with specific ways in which the desire is fulfilled. How sensory representations are generated is addressed in dynamic property ALP4.

ALP4 Generating a belief for the human's sensory representation
If AA believes that the human senses body state,
then it will generate a belief that after Δt the human will have a sensory representation for this.
has_state(AA, belief(has_state(human, sensor_state(B, V, t))))
→ has_state(AA, belief(has_state(human, srs(B, V, t+Δt))))

A person's desire originates from the sensory representation of the body state unbalance. The ambient agent generates a belief on the human's desires by:

ALP5 Generating a belief for the human's desires
If AA believes that the human has a sensory representation for body state *b*
then it will generate a belief that after Δt the human will generate a desire
has_state(AA, belief(has_state(human, srs(b, V, t))))
→ has_state(AA, belief(has_state(human, desire(b, V, t+Δt))))

Next it is shown how the ambient agent estimates the preparations that are triggered. It is assumed that within the person activation of a desire, together with a feeling, induces preparations for a number of action options: those actions the person considers relevant options to satisfy the desire, for example based on earlier experiences. Property ALP6 describes such responses to an activated desire in the form of the preparation for specific actions. It combines the activation levels *V* and V_i of two states (desire and feeling) through connection strengths ω_{1i} and ω_{2i}

respectively. This specifies part of the recursive as-if loop between feeling and body state. This dynamic property uses a combination model based on a function $g(\sigma, \tau, V, V_i, \omega_{1i}, \omega_{2i})$ which includes a sigmoid threshold function

$$th(\sigma, \tau, V) = \frac{1}{1+e^{-\sigma(V-\tau)}}$$

with steepness σ and threshold τ. For this model $g(\sigma, \tau, V, V_i, \omega_{1i}, \omega_{2i})$ is defined as $g(\sigma, \tau, V, V_i, \omega_{1i}, \omega_{2i}) = th(\sigma, \tau, \omega_{1i}V + \omega_{2i}V_i)$ with V, V_i activation levels and ω_{1i}, ω_{2i} weights of the connections to the preparation state.

ALP6 Generating a belief for the human's preparations

If AA believes that the human has a desire for b with level V
and AA believes that the human has feeling B_i with level V_i
and AA believes that the preparation for body state B_i has level U_i
and ω_{1i} is the strength of the connection from desire for b to preparation for B_i
and ω_{2i} is the strength of the connection from feeling of B_i to preparation for B_i
and σ_i is the steepness value for the preparation for B_i
and τ_i is the threshold value for the preparation for B_i
and γ_1 is the person's flexibility for bodily responses
then ambient agent AA will generate the belief that the human's preparation state for body state b_i will occur with level $U_i + \gamma_1(g(\sigma_i, \tau_i, V, V_i, \omega_{1i}, \omega_{2i}) - U_i)\ \Delta t$

has_state(AA, belief(has_state(human, desire(b, V, t)))) &
has_state(AA, belief(has_state(human, feeling(B, V_i, t)))) &
has_state(AA, belief(has_state(human, prep_state(B_i, U_i, t)))) &
has_steepness(prep_state(B_i), σ_i) & has_threshold(prep_state(B_i), τ_i)
→ has_state(AA, belief(
has_state(human, prep_state(B_i, $U_i + \gamma_1 (g(\sigma_i, \tau_i, V, V_i, \omega_{1i}, \omega_{2i}) - U_i) \Delta t$), t+Δt)))

Variants of this property have been used to incorporate interventions which affect the preparations of some B_i : they are assumed to become *0* (suggestion not to do) or *1* (suggestion to do); for example:

has_state(AA, belief(has_state(human, desire(b, V, t)))) &
has_state(AA, belief(has_state(human, feeling(B_i, V_i, t))))
has_state(AA, belief(has_state(human, prep_state(B_i, U_i)))) &
has_state(human, sensor_state(suggestion(do, B_i))))) &
has_steepness(prep_state(B_i), σ_i) & has_threshold(prep_state(B_i), τ_i)
→ has_state(AA, belief(has_state(human, prep_state(B_i, 1, t+Δt))))

The following five properties describe how the ambient agent reasons about the human's body loop.

ALP7 Generating a belief for the human's sensory representation of body states

If AA believes that the human's preparation state for body state B_i with level V_1 occurred
and AA believes that the human senses his body state B_i with level V_2
and AA believes that the human has sensory representation for B_i with level U
and σ is the steepness value for the sensory representation for B_i
and τ is the threshold value for the sensory representation for B_i
and γ is the person's flexibility for bodily responses
then ambient agent AA will generate the belief that the human's sensory representation for body state B_i will occur with level $U + \gamma_2 (g(\sigma, \tau, V_1, V_2, 1, 1) - U)\ \Delta t$.

has_state(AA, belief(has_state(human, prep_state(B_i, V_1, t)))) &

has_state(AA, belief(has_state(human, sensor_state(B_i, V_2, t))))
has_state(AA, belief(has_state(human, srs(B_i, U, t)))) &
has_steepness(srs(B_i), σ) & has_threshold(srs(B_i), τ)
$\rightarrow$ has_state(AA, belief(has_state(human, srs((B_i, U + γ_2 (g(σ, τ, V_1, V_2, 1, 1) - U) Δt), t+Δt)

ALP8 Generating a belief for the human's feelings

If AA believes that the human has a sensory representation for body state B_i with level V,
then it will believe that the human has feeling B_i with level V.

has_state(AA, belief(has_state(human, srs(B_i, V, t)))) $\rightarrow$ has_state(AA, belief(has_state(human, feeling(B_i, V, t+Δt))))

Temporal relationships ALP9, ALP10 and ALP11 below describe the ambient agent's reasoning about how preparations of body states b_i and affect body states b and b_i. The idea is that the actions performed by body states b_i are different means to satisfy the desire related to b, by having an impact on the body state that decreases the activation level V (indicating the extent of unbalance) of body state b. In addition, when performed, each of them involves an effect on a specific body state b_i which can be interpreted as a basis for a form of satisfaction felt for the specific way in which b was satisfied. So, on the one hand a specific action performance involving b_i has an effect on body state b, by decreasing the level of unbalance entailed by b, and on the other hand it has an effect on the body state b_i by increasing the level of satisfaction entailed by b_i. This level of satisfaction may be proportional to the extent to which the unbalance is reduced, but may also be disproportional.

As the possible actions to fulfill a desire are considered different, they differ in the extents of their effects on these two types of body states, according to an effectiveness rate α_i between 0 and 1 for b, and an effectiveness rate β_i between 0 and 1 for b_i. The effectiveness rates α_i and β_i can be considered a kind of connection strengths from the effector state to the body states b and b_i, respectively. In common situations for each action these two rates may be equal (i.e., $\alpha_i = \beta_i$), but especially in more pathological cases they may also have different values where the satisfaction felt based on rate β_i for b_i may be disproportionally higher or lower in comparison to the effect on b based on rate α_i (i.e., $\beta_i > \alpha_i$ or $\beta_i < \alpha_i$). An example of this situation would be a case of addiction for one of the actions. To express the extent of disproportionality between β_i and α_i, a parameter λ_i, called *satisfaction disproportion rate*, between -1 and 1 is used. This parameter relates β_i to α_i using a function f, by $\beta_i = f(\lambda_i, \alpha_i)$. Here the function $f(\lambda, \alpha)$ satisfies $f(0, \alpha) = \alpha$, $f(-1, \alpha) = 0$, and $f(1, \alpha) = 1$. The function $f(\lambda, \alpha)$ can be defined in a continuous (but not differentiable) manner as a piecewise linear function in λ by $f(\lambda, \alpha) = \alpha + \lambda(1-\alpha)$ if $\lambda \geq 0$, and $f(\lambda, \alpha) = (1+\lambda)\alpha$ if $\lambda \leq 0$. Using such f, for normal cases $\lambda_i = 0$ is taken, for cases were satisfaction is disproportionally higher $0 < \lambda_i \leq 1$ and for cases where satisfaction is disproportionally lower $-1 \leq \lambda_i < 0$.

ALP9 Generating a belief for the human's body modification

If AA believes that the human's preparation state for body state B_i with level V occurred,
then it will believe that the human's body state B_i is modified with level V.

has_state(AA, belief(has_state(human, prep_state(B_i, V, t))))
$\rightarrow$ has_state(AA, belief(has_state(human, effector_state(B_i, V, t+Δt))))

ALP10 Generating a belief for the human from effector state to modified body state b_i
If AA believes that the human's body B_i is modified with level V_i,
and AA believes that for each i the effectivity of B_i for b is α_i
and AA believes that the satisfaction disproportion rate of B_i for b is λ_i
then AA will believe that the human's body state B_i will have level $f(\lambda_i, \alpha_i)V_i$.

has_state(AA, belief(has_state(human, effector_state(B_i, V_i, t)))) &
has_state(AA, belief(is_effectivity_for(α_i, B_i, b))) &
has_state(AA, belief(is_disproportion_rate_for(λ_i, B_i)))
→ has_state(AA, belief(has_state(human, body_state(B_i, f(λ_i, α_i)V_i), t+Δt)))

ALP11 Generating a belief for the human from effector state to modified body state b
If AA believes that the human's body B_i is modified with level V_i,
and AA believes that human's body state b has level V,
and AA believes that for each i the effectivity of B_i for b is α_i
then AA believes that human's body state b will have
level $V + (\vartheta * (1\text{-}V) - \rho * (1 - ((1 - \alpha_1 * V_1) * (1 - \alpha_2 * V_2) * (1 - \alpha_3 * V_3))) * V) \Delta t$.

has_state(AA, belief(has_state(human, effector_state(B_i, V_i, t)))) &
has_state(AA, belief(has_state(human, body_state(b, V, t)))) &
has_state(AA, belief(is_effectivity_for(α_i, B_i , b)))
→ has_state(AA, belief(has_state(human, body_state(b,
$V + (\vartheta * (1\text{-}V) - \rho * (1 - ((1 - \alpha_1 * V_1) * (1 - \alpha_2 * V_2) * (1 - \alpha_3 * V_3))) * V) \Delta t)$, t+Δt))

Note that in case only one action is performed (i.e., $V_j = 0$ for all $j \neq i$), the formula in ALP11 above reduces to $V + (\vartheta * (1\text{-}V) - \rho * \alpha_i * V_i * V) \Delta t$. In the formula ϑ is a rate of developing unbalance over time (for example, getting hungry), and ρ a rate of compensating for this unbalance. Note that the specific formula used here to adapt the level of b is meant as just an example. As no assumptions on body state b are made, this formula is meant as a stand-in for more realistic formulae that could be used for specific body states b. A variant of this property has been used to incorporate external events p that incidentally increases the level of the body state (such as exercising):

has_state(AA, belief(has_state(human, effector_state(B_i, V_i, t)))) &
has_state(AA, belief(has_state(human, body_state(b, V, t)))) &
has_state(AA, belief(is_effectivity_for(α_i, B_i , b))) & external_effect(p)
→ has_state(AA, belief(has_state(human, body_state(b,
$V + ((\vartheta + p) * (1\text{-}V) - \rho * (1 - ((1 - \alpha_1 * V_1) * (1 - \alpha_2 * V_2) * (1 - \alpha_3 * V_3))) * V) \Delta t)$, t+Δt))

The strengths ω_{2i} of the connections from feeling b_i to preparation of b_i are considered to be subjected to learning. When an action involving b_i is performed and leads to a strong effect on b_i, by Hebbian learning [3, 15] this increases the strength of this connection. This is an adaptive mechanism that models how experiences in the past may have their effect on behavioural choices made in the future, as also described in Damasio's Somatic Marker Hypothesis [7, 8]. Within the model the strength ω_{2i} of the connection from feeling to preparation is adapted using the following Hebbian learning rule. It takes into account a maximal connection strength 1, a learning rate η, and an extinction rate ζ. A similar Hebbian learning rule can be found in [13]. The agent AA generates beliefs about the connection strengths based on Hebbian learning:

ALP12 Generating a belief for the human's Hebbian learning
If AA believes that the connection from feeling B_i to preparation of B_i has strength ω_{2i}
and AA believes that human has feeling B_i with level V_{1i}

and AA believes that the human's preparation of B_i has level V_{2i}
and the learning rate from feeling B_i to preparation of B_i is η
and the extinction rate from feeling B_i to preparation of B_i is ζ
then after Δt AA will believe that the connection from feeling B_i to preparation of B_i will have strength $\omega_{2i} + (\eta V_{1i} V_{2i} (1 - \omega_{2i}) - \zeta\omega_{2i}) \Delta t$.

has_state(AA, belief(has_connection_strength(feeling(B_i), preparation(B_i), ω_{2i}, t))) &
has_state(AA, belief(has_state(human, feeling(B_i, V_{1i}, t)))) &
has_state(AA, belief(has_state(human, prep_state(B_i, V_{2i}, t)))) &
has_learning_rate(feeling(B_i), preparation(B_i), η) &
has_extinction_rate(feeling(B_i), preparation(B_i), ζ)
→ has_state(AA, belief(has_connection_strength(
feeling(B_i), preparation(B_i), $\omega_{2i} + (\eta V_{1i} V_{2i} (1 - \omega_{2i}) - \zeta\omega_{2i}) \Delta t$)), t+Δt)

Based on the beliefs about the human's states an assessment is made on the level of desire, as follows (where, for example $th = 0.7$):

ALP13 Assessment generation
If AA believes that the human has a desire at time t with level V higher than threshold $th1$,
then an assessment will be generated by AA that human will have a high desire of b at time t

has_state(AA, belief(has_state(human, desire(b, V, t)))) & V ≥ th1
→ has_state(AA, assessment(has_state(human, high_desire(b), t)))

The desire assessment is used to generate an intervention intention, whenever needed. This intention persists until the point in time at which the intervention has to be performed. Table 1 shows the criteria used in the agent's decision process.

Table 1 Assessment criteria used by the ambient agent

	Preparation state level > 0.1	Preparation state level ≤ 0.1
Effectivity rate > 0.5	A good option considered by the human	A good option not considered by the human
Effectivity rate ≤ 0.5	A bad option considered by the human	A bad option not considered by the human

Here the human is assumed to consider an option if the level of the associated preparation state is predicted above a certain threshold, which in the example scenario is set to 0.1, whereas the different options that are available are characterized as good or bad based on the values of the effectivity rates of those options higher or lower than 0.5. Properties ALP14a – ALP17 below accomplish this intervention strategy:

ALP14a Generation of intended intervention by ambient agent: positive suggestion
If AA has generated an assessment that human will have a high desire for b at time t
and AA has desire of human's wellbeing
and AA believes that the human's preparation of B_i has level V_i
and AA believes that for each i the effectivity of B_i for b is α_i
and $V_i < 0.1$ and $\alpha_i > 0.5$
then AA will intends to intervene the human at a later time t to suggest for doing B_i

has_state(AA, assessment(has_state(human, high_desire(b), t))) & has_state(AA, desire(wellbeing(human))) &

has_state(AA, belief(has_state(human, prep_state(B_i, V_i, t+20)))) & $V_i < 0.1$ &
has_state(AA, belief(is_effectivity_for(α_i, B_i, b))) & $\alpha_i > 0.5$
→ has_state(AA, intended_intervention_at(suggestion(human, do, B_i), t))

ALP14b Generation of intended intervention by ambient agent: negative suggestion
If AA has generated an assessment that human will have a high desire for *b* at time *t*
and AA has desire of human's wellbeing
and AA believes that the human's preparation of B_i has level V_i
and AA believes that for each *i* the effectivity of B_i for *b* is α_i
and $V_i > 0.1$ and $\alpha_i < 0.5$
then AA will intends to intervene the human at a later time *t* to suggest for not doing B_i
has_state(AA, assessment(has_state(human, high_desire(b), t))) & has_state(AA, desire(wellbeing(human))) &
has_state(AA, belief(has_state(human, prep_state(B_i, V_i, t+20)))) & $V_i > 0.1$ &
has_state(AA, belief(is_effectivity_for(α_i, B_i, b))) & $\alpha_i < 0.5$
→ has_state(AA, intended_intervention_at(suggestion(human, don't_do, B_i), t))

ALP15 Propagation of intended intervention by ambient agent
If AA intends to intervene the human at a later time *t1* to suggest for eating
and the current time is *t2* and t2 < t1
then the intended intervention by AA will persist
has_state(AA, intended_intervention_at(X, t1)) & current_time(t2) & t2 < t1 → has_state(AA, intended_intervention_at(X, t1))

Finally the intervention is performed:

ALP16 Intervention by Ambient Agent
If AA intends to intervene the human at a later time *t1* to suggest for eating
and the current time is *t2*
and t2 = t1-3
and AA does not observes the human in eating
then AA will suggest the human to eat
has_state(AA, intended_intervention_at(X, t1)) & current_time(t2) & t2 = t1 – 3 → has_state(AA, performed(X))

ALP17 Human sensing of the action performed by agent
If AA suggests human to do X
then the human will sense this
has_state(AA, performed(suggestion(human, X, B)))
→ has_state(human, sensor_state(has_state(AA, performed(suggestion(human, X, B)))))

4 Simulation Results for the Ambient Agent Model

A number of simulations have been performed within the LEADSTO simulation environment [6]. The model was tested in a small scenario, involving an ambient agent and a human, indicated by AA and human, respectively. The example scenario taken here considers a person who is getting hungry which generates a desire to eat for which a number of options is available at that time. As the level of desire increases this makes the person more tempted to eat, and in particular to choose the option that is associated to the best feeling. As the domain model is integrated within the ambient agent, it can predict the human's desire level well in advance, and assesses the extent to which the human will consider the different options that are available to fulfill this desire.

Based on the criteria given in Table 1 above, if the ambient agent predicts that the human will consider those options that are not effective for fulfilling the desire, then it will suggest not to choose them. Similarly, if the assessment process of the ambient agent determines any options that are quite effective for the human to choose, but the human will not consider those, then it will suggest the human to choose them. The scenario starts with some initial values of the human's desire and feeling levels, and then keeps on updating this, using the integrated model explained in Section 3. An example simulation trace (under fixed parameter settings) is illustrated in Fig 1 and 2 (here the time delays within the temporal LEADSTO relations were taken 1 time unit).

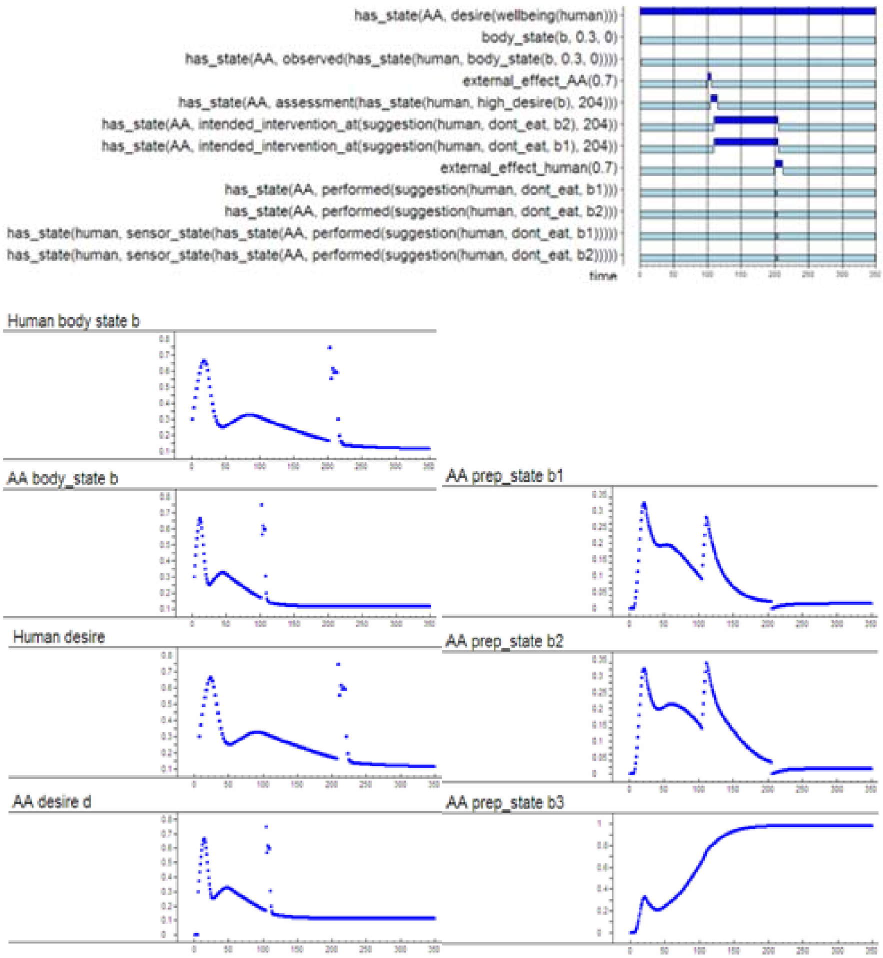


Fig. 1 Simulation Trace 1 – Desire and preparation states ($\alpha_1=\beta_1=0.05$, $\alpha_2=\beta_2=0.25$, $\alpha_3=\beta_3=1$, $\gamma_1=\gamma_2=0.05$, $\sigma_1=\sigma_2=10$, $\tau_1=\tau_2=0.5$, $\rho=0.8$, $\vartheta=0.1$, $\eta=0.04$, $\zeta=0.01$)

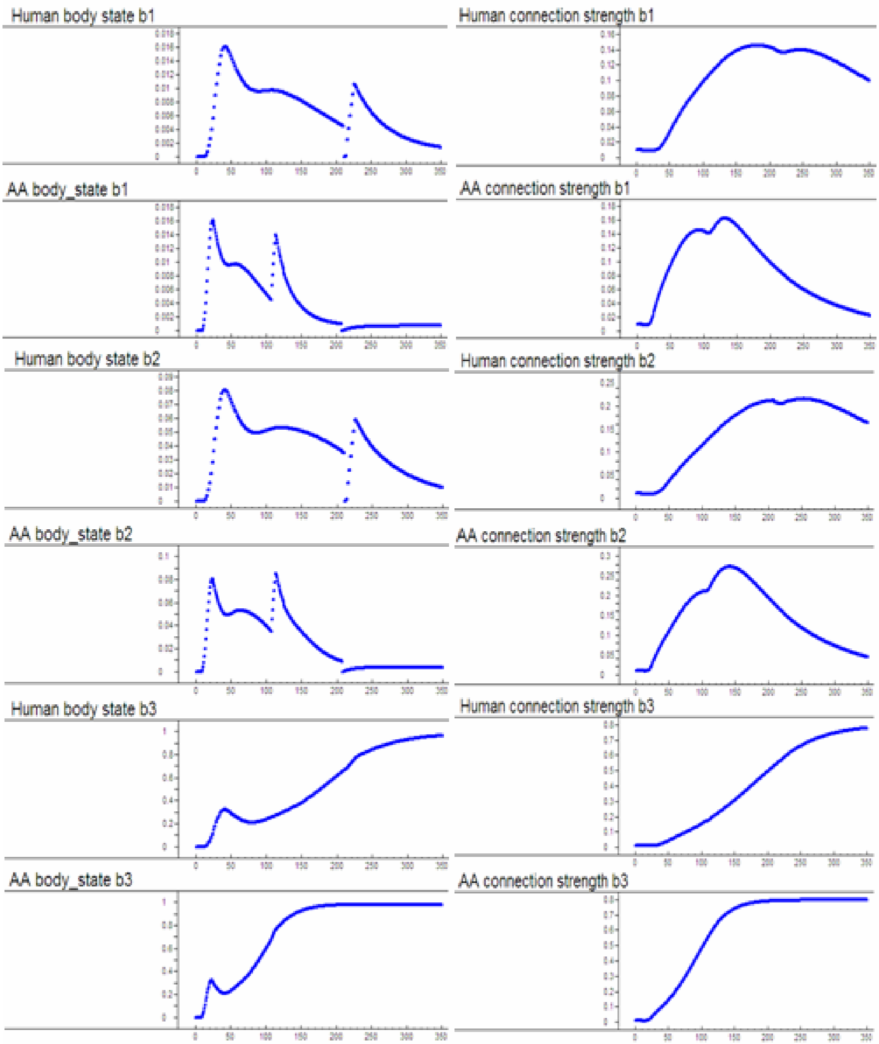


Fig. 2 Simulation Trace 1 – Normal behavior: adaptation process

In these figures, where time is on the horizontal axis, the upper part shows the time periods, in which the binary logical state properties hold (indicated by the dark lines); for example, has_state(AA, assessment(has_state(human, high_desire(b), 204))). Below this part, quantitative information is provided about the human's actual desire, preparation states, connection strength levels, levels of different body states and the ambient agent AA's prediction of these. Values for these levels for the different time periods are shown by the dark lines. Note that the scale on the vertical axis differs over the different graphs, and only a selection of the relevant state properties is shown.

For the example trace shown in Fig 1 and 2, for each i that represents an option, $\lambda_i = 0$ was taken, so in this example simulation the human is not developing an addiction to any option. Option 3 has the highest effectiveness rate, i.e. $\alpha_3 = 1$. Its value is substantially higher than the rates for the other two available options. This affects the respective body states. Furthermore, as can be seen in Fig 2 by the Hebbian learning it gives a strong effect on the strength of the connection from feeling to preparation for this option: the connection strength for option 3 increases over time until it reaches an equilibrium state.

As shown in the lower part of the Fig 2, at time point 10, the ambient agent predicts that the desire level of human will increase but it will not cross the threshold set to *0.7*, i.e., it is not considered sufficient enough to make the human tempted to choose this option. This is confirmed by the graph of the desire level of the human, where at time point 20, it increases but does not cross the threshold. Hence the ambient agent does not intend to perform any action. But later, some external effects (e.g., the human's habit to attend gym) causes an increase in this desire level, which is predicted by the ambient agent AA in the simulation at time point 102, as shown in the upper part of the Fig 1, by the state property has_state(AA, assessment(has_state(human, high_desire(b), 204))), expressing that an assessment has been generated that the human will have a high desire for b at time 204. Thereafter, as described in Table 1, AA predicts that the human will choose all three options because of the high values of the preparation states for those options, as shown in Fig 1, in the graph of AA prep_state b1, AA prep_state b2 and AA prep_state b3. After this, the agent will assess for these options whether they are good or bad, based on their effectivity rates. For this particular example simulation, the options *b1* and *b2* are assessed as bad because of their low effectivity rates, i.e., $\alpha_1 = 0.05$, $\alpha_2 = 0.25$, which are lower than the threshold set to *0.5*. On the other hand, option *b3* is assessed as good because its effectivity rate is higher than threshold, i.e., $\alpha_3 = 1$. Hence the ambient agent generates the intention to suggest the human not to choose options *b1* and *b2* as shown in the upper part of Fig 1, by the state property has_state(AA, intended_intervention_at(suggestion(human, don't_eat, b1), 204)) and similarly for *b2*.

5 Analysis of the Ambient Agent Model by Automated Verification

In order to investigate whether the ambient agent indeed acts according to what is expected, some logical properties (requirements) have been identified, formalised, and verified against the simulation traces of the model. In this section, first the language used to express such properties is briefly introduced, followed by the specification of the actual properties, and the result of their verification. Using a formal specification for desired properties of the ambient agent enables automatic verification of them against simulation traces. This automated verification is performed using the logical language TTL and its software environment [5]. The temporal predicate logical language TTL supports formal specification and analysis of dynamic properties, covering both qualitative and quantitative aspects. TTL is built on atoms referring to *states* of the world, *time points* and *traces*, i.e. trajectories of states over time. *Dynamic properties* are temporal statements formulated with respect to traces based on the state ontology Ont in the following manner. Given a trace γ over state ontology Ont, the state in γ at time point t is denoted by state(γ, t). These states are related to state properties via the infix predicate |=, where state(γ, t) |= p denotes that state property p holds in trace γ at time t. Based on these statements, dynamic properties are formulated in a sorted first-order predicate logic, using quantifiers over time and traces and the usual first-order logical connectives such as ¬, ∧, ∨, ⇒, ∀, ∃. For more details, see [5]. An overall property to be satisfied by the agent is that if the level of a desire of the human exceeds a particular threshold, it should eventually become below the threshold.

P1(th:real): Successful support
In all traces γ and all time points t the level of a desire of the human if the desire of a human exceeds the threshold, then there exists a later time point at which this is not the case.
∀t:TIME, γ:TRACE, V:REAL
[state(γ, t) |= desire(b, V) & V > th ⇒ ∃t2:TIME, V2:REAL [state(γ, t2) |= desire(b, V2) & V2 < th]]

For the simulation traces generated using the ambient agent model, this property is satisfied for all traces (with a threshold value of *0.7*). The overall behavior as expressed in P1 can be accomplished by intervention by giving one or more suggestions at the right moment (expressed in P2) in combination with the human responding to these suggestions (expressed in P3).

P2(th:real, d:duration): Right moment for intervention
In all traces γ, if the ambient agent at time point t1 predicts that at time point t2 the human will have a desire exceeding the threshold th, then the ambient agent will give a suggestion to the human.

∀t1, t2:TIME, γ:TRACE, V:REAL
[[state(γ, t1) |= has_state(AA, belief(has_state(human, desire(b, V, t2)))) & V > th]
⇒ ∃t3:TIME > t1:TIME, A:ACTION, B:BODY_STATE
[state(γ, t3) |= has_state(AA, performed(suggestion(human, A, B)))]]

This property holds for all traces (when a threshold of *0.7* is chosen).

P3(d:duration): Right response
In all traces γ, if the ambient agent gives a suggestion to the human at time point t to either avoid a body state *B* (don't eat for this case) or accomplish a body state *B* (i.e., eat), then the human will follow this suggestion, indicated by a preparation state for *B* being *0* for the case of an avoidance suggestion, or a *1* in case of an accomplish suggestion for the body state *B*.
∀t1:TIME, γ:TRACE, V:REAL, B:BODY_STATE
[[state(γ, t1) |= has_state(AA, performed(suggestion(human, dont_do, B))) ⇒
∃t2:TIME > t1 [state(γ, t2) |= prep_state(B, 0)]] &
[state(γ, t1) |= has_state(AA, performed(suggestion(human, do, B))) ⇒
∃t2:TIME > t1 [state(γ, t2) |= prep_state(B, 1)]]

This last property is satisfied for all traces as well.

6 Discussion

To function in a knowledgeable manner, ambient agents [1, 18] need a model of the humans they are supporting. Such a model enables them to perform a form of mindreading [12, 14]. The ambient agent model presented here focuses on mindreading concerning the interaction between desires, preparations and feelings, based on neurological theories that address this interaction. The integrated dynamical model describes more specifically how a desire induces (as a response) a set of preparations for a number of possible actions, involving certain body states, which each affect sensory representations of the body states involved and thus provide associated feelings. On their turn these feelings affect the preparations, for example, by amplifying them. In this way an agent model is obtained for desiring which integrates both cognitive and affective aspects of mental functioning. For the interaction between feeling and preparation of responses, a converging recursive body loop is included in the dynamical model, based on elements taken from [9, 10]. Both the strength of the preparation and of the feeling emerge as a result of the dynamic pattern generated by this loop. The dynamical model is adaptive in the sense that within these loops the connection strengths from feelings to preparations are adapted over time by Hebbian learning [3, 13, 15]. By this adaptation mechanism, in principle the person achieves that the most effective action to fulfill a desire is chosen. However, the dynamical model can also be used to cover humans for whom satisfaction for an action is not in proportion with the fulfillment of the desire, as occurs, for example, in certain cases of earlier addictive experiences which provide temptations for the future. In this case, action choice may become biased by such temptations, and this is where an ambient agent can play a supporting role. The agent model equipped with the dynamical model for the dynamics of desires, preparations and feelings was specified in the hybrid dynamic modelling language LEADSTO, and simulations were performed in its software environment [6]. Simulation experiments show that the model behaves as expected, which also have been verified formally.

References

1. Aarts, E., Collier, R., van Loenen, E., de Ruyter, B. (eds.): EUSAI 2003. LNCS, vol. 2875. Springer, Heidelberg (2003)
2. Bechara, A., Damasio, A.: The Somatic Marker Hypothesis: a neural theory of economic decision. Games and Economic Behavior 52, 336–372 (2004)
3. Bi, G.Q., Poo, M.M.: Synaptic Modifications by Correlated Activity: Hebb's Postulate Revisited. Ann. Rev. Neurosci. 24, 139–166 (2001)
4. Bosse, T., Hoogendoorn, M., Klein, M., Treur, J.: A Generic Agent Architecture for Human-Aware Ambient Computing. In: Mangina, E., et al. (eds.) Agent-Based Ubiquitous Computing. Ambient and Pervasive Intelligence, vol. 1, pp. 35–62. World Scientific Publishers/Atlantis Press (2009)
5. Bosse, T., Jonker, C.M., van der Meij, L., Sharpanskykh, A., Treur, J.: Specification and Verification of Dynamics in Agent Models. Int. Journal of Cooperative Information Systems 18, 167–193 (2009)
6. Bosse, T., Jonker, C.M., van der Meij, L., Treur, J.: A Language and Environment for Analysis of Dynamics by Simulation. International Journal of Artificial Intelligence Tools 16, 435–464 (2007)

7. Damasio, A.: Descartes' Error: Emotion, Reason and the Human Brain. Papermac, London (1994)
8. Damasio, A.: The Somatic Marker Hypothesis and the Possible Functions of the Prefrontal Cortex. Philosophical Trans. of the Royal Society: Biological Sciences 351, 1413–1420 (1996)
9. Damasio, A.: The Feeling of What Happens. Body and Emotion in the Making of Consciousness. Harcourt Brace, New York (1999)
10. Damasio, A.: Looking for Spinoza. Vintage books, London (2004)
11. Eich, E., Kihlstrom, J.F., Bower, G.H., Forgas, J.P., Niedenthal, P.M.: Cognition and Emotion. Oxford University Press, New York (2000)
12. Gärdenfors, P.: How Homo Became Sapiens: On the Evolution of Thinking. Oxford University Press, Oxford (2003)
13. Gerstner, W., Kirstner, W.M.: Mathematical formulations of Hebbian learning. Biol. Cybern. 87, 404–415 (2002)
14. Goldman, A.I.: Simulating Minds: the Philosophy, Psychology and Neuroscience of Mindreading. Oxford University Press, Oxford (2006)
15. Hebb, D.: The Organisation of Behavior. Wiley, New York (1949)
16. Marsella, S.C., Pynadath, D.V., Read, S.J.: PsychSim: Agent-based modeling of social interaction and influence. In: Lovett, M., et al. (eds.) Proc. of the Int. Conf. on Cognitive Modelling, ICCM 2004, pp. 243–248 (2004)
17. Niedenthal, P.M.: Embodying Emotion. Science 316, 1002–1005 (2007)
18. Riva, G., Vatalaro, F., Davide, F., Alcañiz, M. (eds.): Ambient Intelligence. IOS Press, Amsterdam (2005)
19. Winkielman, P., Niedenthal, P.M., Oberman, L.M.: Embodied Perspective on Emotion-Cognition Interactions. In: Pineda, J.A. (ed.) Mirror Neuron Systems: the Role of Mirroring Processes in Social Cognition, pp. 235–257. Springer Science, Heidelberg (2009)

Cluster Analysis and Decision Trees of MR Imaging in Patients Suffering Alzheimer's

Ali Hamou, Michael Bauer, Benoit Lewden, Andrew Simmons, Yi Zhang, Lars-Olof Wahlund, Catherine Tunnard, Iwona Kloszewska, Patrizia Mecozzi, Hilkka Soininen, Magda Tsolaki, Bruno Vellas, Sebastian Muehlboeck, Alan Evans, Per Julin, Niclas Sjögren, Christian Spenger, Simon Lovestone, Femida Gwadry-Sridhar, and other participants in the AddNeuroMed consortium

Ali Hamou · Benoit Lewden · Femida Gwadry-Sridhar
I-THINK Research Lab, Lawson Health Research Institute, London, ON Canada

Michael Bauer
Department of Computer Science, University of Western Ontario, London, ON Canada

Andrew Simmons · Catherine Tunnard · Simon Lovestone
NIHR Specialist Biomedical Research Centre for Mental Health at the South London and Maudsley NHS Foundation Trust and King's College London and the MRC Centre for Neurodegeneration, King's College London, Institute of Psychiatry, De Crespigny Park, London SE5 8AF

Yi Zhang · Sebastian Muehlboeck · Christian Spenger
Department of Clinical Science, Intervention and Technology, Division of Radiology, Karolinska Institutet, Huddinge, Stockholm, Sweden

Lars-Olof Wahlund
Department of Neurobiology, Care Sciences and Society, Section of Clinical Geriatrics, Karolinska Institutet, Karolinska University Hospital, Huddinge, Stockholm, Sweden

Iwona Kloszewska
Department of Old Age Psychiatry & Psychotic Disorders, Medical University of Lodz, 92-216 Lodz, Poland

Patrizia Mecozzi
Section of Gerontology and Geriatrics, Department of Clinical and Experimental Medicine, University of Perugia, Perugia, Italy

Hilkka Soininen
Department of Neurology, Kuopio University and University Hospital, PO Box 1777, Kuopio, Finland

Magda Tsolaki
Department of Neurology, Thessaloniki, Greece

Bruno Vellas
Department of Internal and Geriatrics Medicine, Hôpitaux de Toulouse, Toulouse, France

Alan Evans
Montreal Neurological Institute, McGill University, Montreal, Canada

Per Julin · Niclas Sjögren
AstraZeneca R&D Södertälje, Sweden

Other participants in the AddNeuroMed consortium making essential contributions to this work are listed in supplementary data

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 477–484.
springerlink.com

Abstract. The use of novel analytical techniques (such as data clustering and decision trees) that can model and predict patient disease outcomes has great potential for assessing disease process and progression in Alzheimer's disease and mild cognitive impairment. For this study, 43 different variables (generated from image data, demographics and clinical data) have been compiled and analyzed using a modified clustering algorithm. Our aim was to determine the influence of these variables on the incidence of Alzheimer's and mild cognitive impairment. Furthermore, we used a decision tree algorithm to model the level of "importance" of variants influencing this decision.

Keywords: Alzheimer's Disease, Mild Cognitive Impairment, Cluster Analysis, Decision Tree Analysis, MRI.

1 Introduction

Mild cognitive impairment (MCI) may be defined as a transitional state between normal aging and *Alzheimer's disease* (AD) in which memory impairment is greater than expected for age, but general cognitive function and daily living activities are preserved [1]. The rate of progression from MCI to dementia is considerable according to estimates from several longitudinal studies: 12% per year for 4 years [1], 40% over 2 years [2], 53% over 3 years [3], 34%–100% over 4–5 years [4,5], and 100% over 9.5 years [6]. Currently, there is much debate over whether MCI is a separate nosological entity at increased risk of dementia or is an early symptom of AD [7]. Regardless, it is important to develop tools that enable us to predict conversion from MCI into progressive dementia.

The early detection and management of this disorder is important due to the recent availability of drug therapies for mild-to-moderate AD. As therapies with the potential to arrest the progression of Alzheimer's disease are being developed, identification of people with pre-Alzheimer conditions has become increasingly important to determine whether such individuals can benefit from pharmacologic and non-pharmacologic interventions.

Unfortunately, once fully developed, AD is a devastating illness with a long-drawn-out course that results in a diminished quality of life for patients and their caregivers. Treatment strategies that are able to detect the most susceptible individuals (either primary or secondary prevention) are urgently needed. These strategies can provide a means of delaying the progression of the disease to those that are at risk of converting from MCI to AD. Developing analytic tools to support these goals is in the interest of the patient, health care provider and payers, as well as research and development in the pharmaceutical industry. In order to refine treatment strategies and establish early markers of disease, it is important to differentiate features that are prevalent in AD and MCI.

This work deals with the use of modeling techniques to classify patients as AD and MCI, which will improve clinical outcomes by providing an early assessment of potential AD patients. This assessment can change the treatment plans of such patients, which could impact the long term progression of AD or MCI. Section 2 describes the systems, methods and approaches used in detail and describes implications of the proposed model. Section 3 describes the analysis used on the datasets and Section 4 concludes and summarizes this work.

2 Systems and Methods

In order to address these goals, data from the AddNeuroMed study[1], (which had institutional review board approval), was used to accurately model AD and MCI. A rich imaging dataset was available, which provided efficacy data and optimized patient selection. Data was collected at baseline (time of recruitment) and then at three month and 12 months follow up visits. In this paper, we describe the methodology used to differentiate patients at baseline.

2.1 Dataset Source

AddNeuroMed, a European AD biomarker study, aimed to perform a multi center MRI study similar to that of a drug trial. MRI scans were collected and evaluated at a data coordination centre (DCC) and were correlated to data from other sources (i.e. clinical datasets, data from blood samples, etc). The clinical centers that participated in this study were Kings Collage London, University of Thessaloniki, University of Kuopio, University of Perugia, Medical University of Lodz and University of Toulouse. The DCC was the Karolinska Institutet in Stockholm.

Data management, image quality control and workflow control were guided by the Loris database platform (originally developed at McGill Brain Imaging Centre), and were further customized in this work. Patients were selected based on criteria that established whether they had MCI, AD or neither (i.e. controls). Patients were selected at each of the above centres, and were continuously sampled until the accrual goals were met. Accrued subjects consisted of a balanced sample with 112 healthy control, 122 MCI patients and 120 AD patients.

2.2 Tissue Classification and Regional Segmentation

A highly automated structural MRI analysis system was utilized for data analysis consisting of image intensity non-uniformity correction, segmentation of brain tissue and regional brain parcellation.

Data was initially corrected for intensity non-uniformity using the N3 algorithm [8]. This is a fully automated technique which maximizes the entropy of the intensity histogram.

The images were subsequently segmented into gray matter, white matter, Cerebrospinal fluid (CSF) using an artificial neural network classifier called INSECT (Intensity-Normalized Stereotaxic Environment for Classification of Tissues) which allowed to calculate the total brain volume, the total gray matter, total white matter and total CSF.

[1] This study was funded by the European Union of the Sixth Framework program priority FP6-2004-LIFESCIHEALTH-5, Life Sciences, Genomics and Biotechnology for Health.

Regional separation of the brain tissue was subsequently achieved using a multi-scale analysis ANIMAL technique (Automated Non-linear Image Matching and Anatomical Labeling) [9] which deforms the T1-weighted MP-RAGE volume to match a previously labeled MRI volume. Anatomical labels are defined in the new volume by interpolation from the original labels, via the 3D deformation field. Thus this allowed to calculate for example volumes of brainstem, caudate, cerebellum, fornix, frontal-, parietal-, occipital- gray and white matter, putamen, sub thalamic nucleus, thalamus globus pallidus, and all ventricles separately left and right, additional generating 33 variables.

2.3 Cortical Thickness Measurements

To measure the cortical thickness, a method for segmenting the entire human cerebral cortex from MRI is employed. Cortical thickness is calculated by CLASP [10], which produced the values for the mean cortical thickness of the whole brain and that of the left and right hemispheres. Measuring surface-based cortical thickness with the CLASP-method gives the advantage of providing a direct quantitative index of cortical morphology. Once complete, it is possible to detect differences in cortical thickness in presymptomatic individuals with a familial variant of frontotemporal dementia.

2.4 Variables

The T1 weighted images were used to measure structural changes in the brain. As described previously, different types of image processing were used. Then tissue classification and regional segmentation were done resulting in 36 different volumetric measurements. The cortical thickness provided an additional three variables. This gave a total of 39 MRI variables. Two clinical tests were included, MMSE and the CERAD word list learning record test. Thus, altogether 43 variables were used for the proposed analysis including age and gender. Table 1 depicts some of the demographics of the covariates used for the proposed algorithm.

Table 1 Demographics of AddNeuroMed dataset

Variable	Alzheimer's Disease	Mild Cognitive Impairment	Control
Number	120	122	112
Gender - f/m	77/43	63/59	63/49
Age	75.5 ±6.1	74.4 ±5.7	73.0 ±6.7
MMSE	20.9 ±4.8	26.8 ±3.0	29.0 ±1.2
CDR	1.2 ±0.5	0.5 ±0.1	0
Years of Education	8.1 ± 4.0	8.9 ±4.3	10.7 ±4.8

Data is represented as average ± standard deviation, MMSE = Mini Mental State Examination, CDR = Clinical Dementia Rating (validated measures).

3 Analysis

A cluster analysis algorithm was used in order to determine whether or not partitions, representing AD or MCI states, within the AddNeuroMed dataset could be identified and modeled for subsequent decision making. This technique was further refined using decision trees in order to enhance the cluster predictor.

3.1 Cluster Analysis

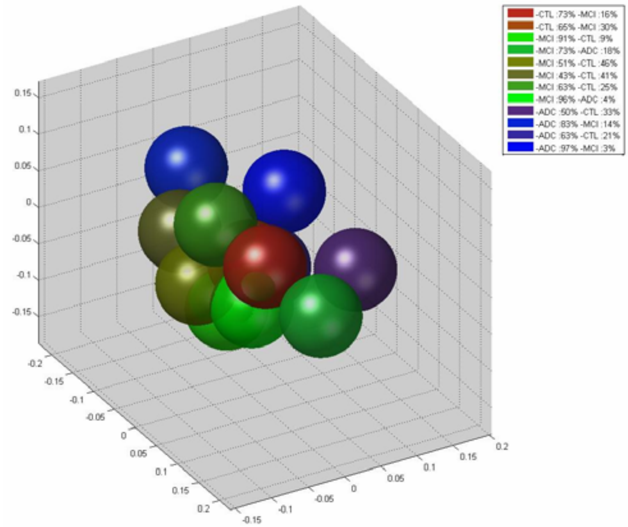
Cluster analysis, or clustering, is the assignment of a set of observations into subsets (called clusters) such that observations in the same cluster are of similar kind and grouped into respective categories. Clustering is a method of unsupervised learning, and a common technique for statistical data analysis. The clustering method used in this work was developed to serve two purposes, the first was to expose the importance of each variable with respect to the actual decision making process on the disease. The second was to provide a means of discovering, given the preliminary image data results, whether or not a subject was likely to have MCI or AD based on cluster belongingness.

A variety of different clustering sizes and algorithms were explored. Empirically, the number of clusters was chosen to be 12 – resulting in clusters that were neither too large to manage (over fitted), nor ones with too few data points to give an adequate determination of patient status due to lack of separation. A modified k-means clustering algorithm was used in this work due to its speed of execution on large datasets. This algorithm incorporates both the internal consistency of variables within the cluster (distance to the cluster's centroid) and the external distance to all neighboring clusters as its heuristic. The k-means algorithm used is refinement iterative and alternates between assignment (where each variable is assigned to a cluster) and updating (calculation of new means of the cluster centre). By doing so, clusters will eventually incorporate all variables within the closest proximity space.

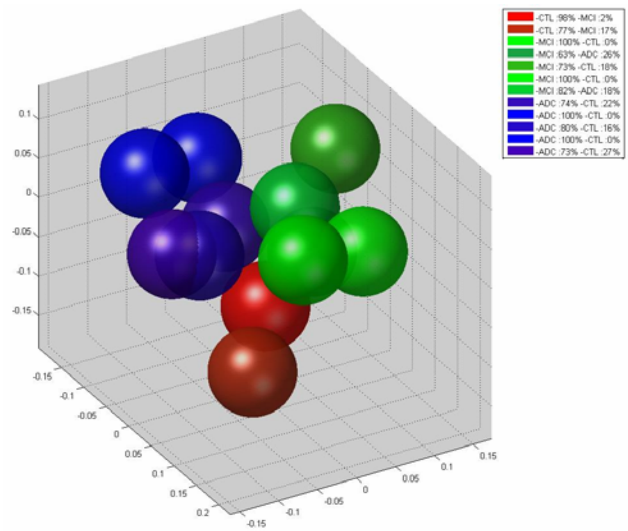
During cluster generation, the algorithm is applied to every variable within the dataset. Variables are then ranked by influence and probability of defining a cluster and its location within the cluster field (internal consistency measure). Following cluster creation, the cluster fields are grouped based on the internal consistency of the variables within each cluster. The mathematical indicator used to create these groupings is defined as the orthogonal distance between clusters.

Since it is known whether a patient belongs to the control, MCI or AD group, this information (which is not used in cluster creation) is used to validate the biological plausibility of the separation, in other words, test the cluster field specificity. Hence, the cluster field can be used in order to determine which variable belongs to a respective group, given an internal consistency measure.

Figure 1(a) displays a cluster field using the proposed technique with belongingness to the groups indicated by colour. This was generated by using all of the variables (43 for AddNeuroMed study) available. This technique leads to nicely separated clusters, however there is overlap between groupings (where primary



(a)



(b)

Fig. 1. Cluster analysis of AddNeuroMed dataset modeled on 3D space proximity grid. (a) Full variable selection to create a cluster fields. (b) Using only the variables determined to be of value by decision trees. (Implementations of clustering and graphing were done using C++ and Matlab).

red represents control, primary green represents MCI, and primary blue represents AD). This is evident in Figure 1(a), since the colours shown have amalgamated at various places within the proximity space. This is known as grouping overlap within the clusters, and hence makes it difficult to accurately define what a cluster truly represents – rendering decision making complicated. Accurate variable selection can lead to better defined clusters with less overlap in groupings. Since the grouping results are known, a series of principal component analysis could be used to refine the variables used. Though, since the ultimate goal of the cluster analysis is decision making, decision trees are a more efficient means of refining this process.

3.2 Decision Trees

In order to determine which variables increase the probability of a resulting cluster belonging to a specific group (i.e. one outcome versus another), a decision tree algorithm is used. Consider the resulting cluster analysis that is rendered in a 3D proximity space (eg. Figure 1(a)) where the colours are distinct and their separation gives rise to a grouping). The legend indicates the percentage of patients with a specific outcome within the group. Groups with greater than 80% of the outcome provide insight into variables that tend to collect or accumulate together within that group. Decision trees allow us to expose the strongest variables within each grouping and hence give more weight to these variables during cluster creation.

For each grouping, a decision tree algorithm was run on the variables. For the tree creation, the variables with the greatest importance, based on external proximity between groups, were used to create the decision tree. The probability at the resulting nodes indicated whether or not that variable influenced a grouping. The variables that had an influence of greater than 80% were allocated greater priority in cluster creation. Figure 1(b) shows a render of the dataset with variable selection based on decision trees, on a 3D space proximity grid. It is apparent that the colours of Figure 1(b) are elemental and hence better separation of stage (MCI vs. AD).

4 Discussion and Conclusions

The utilization of clustering and decision trees has potential to provide models which demonstrate the influence and importance of clinical, biomarker imaging, and neuro-psychological variables on a patient's long term health. This research focused on exploring whether such techniques could be applicable to MRI measures from AD and MCI. Using the proposed techniques, we believe that it will be possible in the future to aid the early detection of AD/MCI patients and allow the application of new drug therapies with the potential to arrest or retard the progression of Alzheimer's disease.

For this study, a modified k-means clustering algorithm was used to partition data into control, MCI and AD sets. Decision tree models were used to identify the level of "importance" of covariates within each cluster. The algorithm provided adequate separation of groups where decision making is possible using this model.

Future improvements and refinements to the clustering algorithm will include the analysis of each covariate, by calculating the distance of each variable to the centre of the cluster. This is one of the first studies to consider clustering in order to help partition variables on the basis of clinical and biological importance. The implications for future drug development and application are supported by our early findings.

References

1. Petersen, R.C., Smith, G.E., Waring, S.C., Ivnik, R.J., Tangalos, E.G., Kokmen, E.: Mild cognitive impairment: clinical characterization and outcome. Arch. Neurol. 56, 303–308 (1999); correction: 56, 760
2. Johnson, K.A., Jones, K., Holman, B.L., Becker, J.A., Spiers, P.A., Satlin, A., Albert, M.S.: Preclinical prediction of Alzheimer's disease using SPECT. Neurology 50, 1563–1571 (1998)
3. McKelvey, R., Bergman, H., Stern, J., Rush, C., Zahirney, G., Chertkow, H.: Lack of prognostic significance of SPECT abnormalities in non-demented elderly subjects with memory loss. Can. J. Neurol. Sci. 26, 23–28 (1999)
4. Bennett, D.A., Wilson, R.S., Schneider, J.A., Evans, D.A., Beckett, L.A., Aggarwal, N.T., Barnes, L.L., Fox, J.H., Bach, J.: Natural history of mild cognitive impairment in older persons. Neurology 59, 198–205 (2002)
5. Krasuski, J.S., Alexander, G.E., Horwitz, B., Daly, E.M., Murphy, D.G., Rapoport, S.I., Schapiro, M.B.: Volumes of medial temporal lobe structures in patients with Alzheimer's disease and mild cognitive impairment (and in healthy controls). Biol. Psychiatry 43, 60–68 (1998)
6. Morris, J.C., Storandt, M., Miller, J.P., McKeel, D.W., Price, J.L., Rubin, E.H., Berg, L.: Mild cognitive impairment represents early-stage Alzheimer disease. Arch. Neurol. 58, 397–405 (2001)
7. Ritchie, K., Touchon, J.: Mild cognitive impairment: conceptual basis and current nosological status. Lancet 355, 225–228 (2000)
8. Sled, J.G., Zijdenbos, A.P., Evans, A.C.: Nonparametric method for automatic correction of intensity nonuniformity in MRI data. IEEE Trans. Med. Imag. 17, 87–97 (1998)
9. Collins, D., Holmes, C., Peters, T., Evans, A.: Automatic 3D model-based neuroanatomical segmentation. Human Brain Mapping 3(3), 190–208 (1995)
10. Kim, J.S., Singh, V., Lee, J.K., Lerch, J., Ad-Dab'bagh, Y., MacDonald, D., Lee, J.M., Kim, S.I., Evans, A.: Automated 3-D extraction and evaluation of the inner and outer cortical surfaces using a Laplacian map and partial volume effect classification. NeuroImage 27(1), 210–221 (2005)

Improving the Quality of Health Service with Smart Communication

Rosa Cano, Manuel P. Rubio, and Ana de Luis

Abstract. This article presents the Hippocrates of Cos Multi-Agent System 1.0 (HC-MAS 1.0) a multi-agent system (MAS) based on PAINALLI, an open MAS architecture based on virtual organizations. PAINALLI utilizes an agent-based model to facilitate the development of MAS and optimize the communication process in business organizations. The architecture and the system can access information and services ubiquitously from either land or mobile devices connected to wired or wireless networks. HC-MAS 1.0 supports the communication process for the healthcare services of a hospital. This paper presents the results obtained from the implementation of the system and demonstrates the advantages produced by the use of this new technology.

Keywords: Quality, medical informatics, HCE, HL7, DICOM, ISO 9000 y 15489, organizational communication, open MAS and virtual organizations.

1 Introduction

Communication is one of the fundamental processes in which individuals engage throughout their lifetime. It is the tool that society uses to advance towards the compliance of the demands and needs of its members. As with specific societies, business organizations require the tools, systems and models that allow them to

Rosa Cano
Departamento de Sistemas e Informática, Instituto Tecnológico de Colima,
Av. Tecnológico, 1, 28976, Villa de Álvarez, Colima, México

Manuel P. Rubio
Departamento de Informática y Automática, Universidad de Salamanca,
Escuela Politécnica Superior de Zamora
Av. de Requejo, 33, 49022 Zamora, Spain

Ana de Luis
Departamento de Informática y Automática, Universidad de Salamanca.
Plaza de la Merced s/n, 37008, Salamanca, Spain
e-mail: {rcano,mprc,adeluis}@usal.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 485–492.
springerlink.com

adapt to the new knowledge-based society on a daily basis. They need to adapt their strategies in order to become the type of business organization that learns and focuses on teamwork, the optimal use of human resources, flexibility, and the involvement of professionals in an innovative corporate culture that is fully integrated within a society. Such organizations emphasize just-in-time objectives, quality, efficiency and the continual improvement of processes [1].

Among the most prominent MAS objectives is the ability to build systems capable of autonomous and flexible decision-making, and to cooperate with other systems within a ¨society¨ that meets the following requirements: distribution, continual evolution, flexibility to allow members (agents) to enter and exit, the appropriate management of an organizational structural, and the ability to execute agents in both multi devices and other devices with limited resources. It is possible to meet these requirements with the use of an open system and virtual organization paradigm [3].

Open MAS [4] can be thought of as distributed systems in which the agent population interacts and exhibits various behaviors. Unlike distributed systems, the cooperation between agents is not determined during the design process, rather it emerges upon execution.

The objective of this paper is to present the HC-MAS 1.0 system, an open MAS that can manage the communication process within healthcare environments. The system is based on the PAINALLI architecture, which is used for developing open MAS for virtual organizations. This makes it easier to develop open MAS systems that can optimize the communication process in business organizations by using an agent platform, which can be accessed via fixed and mobile devices such as PDAs (Personal Digital Assistant), smart phones and personal computers connected to wired and wireless networks.

The majority of the methodologies used in the development of MAS are an extension of existing methodologies from other fields [5] . Some are based on specific agents and architectures [6], while others are geared towards organizations, Agent-Group-Role [7], Tropos [8] , MOSEInst [9], OMNI [10], and E-institutions. There does not currently exist a methodology for developing MAS that is based on organizational structures and that would allow direct communication between the various communicative aspects within an organization. Nor is there a methodology that allows a complete specification of the system´s social structure. PAINALLI is based on THOMAS (MeTHods, Techniques and Tools for Open Multi-Agent Systems) [12], a system that strengthens the definition for the organizational model and the specific methods for the development of open MAS focused on the concept of organization.

Poor internal communication is one of the weaknesses that most affects the development of an organization. For this reason, it is necessary to create a new business model in which communication assumes a critical role [14].

The following section describes the components of the architecture. Section 3 presents the HC-MAS 1.0 system, and section 4 offers the results and conclusions.

2 Painalli Architecture

PAINALLI is an architecture that is used for developing open MAS based on virtual organizations. It focuses specifically on the communication process in business organizations. The communication process in PAINALLI is based on the use of templates and metadata that are associated with attached files. The input methods are based on international standards: ISO 15489 e ISO 9000:2000.

The goals for the architecture were rooted in the exchange of messages and attached files as a support mechanism for an organization´s communication process. The primary objectives are to manage personal and business related messages, documents, user accounts, user groups, and agendas. All of the information is stored and transferred with encryption algorithms.

Users are able to manage their own messages, which are created from templates: scheduling, information, question and request. The sender is notified once the message has been read. A conversation is a set of messages with a common theme, which is ¨owned¨ by the original sender. Only the original sender can eliminate messages and conversations.

The architecture administrator is responsible for managing the templates and the input methods, both of which can be tailored to the specific needs of the organization. The administrator manages the users, who can be grouped according to the organizational structure.

2.1 Architecture Components

PAINALLI is composed of: applications and services, communication protocol, and an agent platform

✓ **Applications and Services**

The combined set of applications and services proposed in PAINALLI constitute the foundation that support the functionalities given to the users and developers who can now benefit from web services such as: web interoperability, offering services through an open exchange of applications and data, offering remote procedure services, and requesting procedure services through the web.

The programs required for accessing the system functionalities are the applications. They must be dynamic and able to adapt to their context, and capable of reacting differently to particular situations and the type of service requested. The applications make it possible for the services to be executed locally or remotely, through computers or mobile devices. The low processing capability of the applications is irrelevant since computing tasks are carried out by the agents and services located in the devices specifically intended for these activities.

Services constitute a set of activities that are meant to respond to the needs of the requester. They are programs that provide methods for accessing databases, manage connections, analyze data, obtain information from external devices, publish information, or even use other services to carry out a particular task.

PAINALLI uses service directories that can be managed dynamically according to the needs of the developers and the demands of the users. Both services and applications in PAINALLI are reusable and function independently from the system to which they are offering their functionalities and from the programming language used by the platform agents.

✓ **Communication Protocol**

Messages are transmitted in PAINALLI through a communication protocol that allows applications and services to communicate directly with the agent platform. The SOAP (*Service Oriented Architecture Protocol*) [15] specification serves as a reference for establishing the communication protocol in PAINALLI and allows the programming language to function independently. Agents use the ACL (*Agent Communication Language*) specification in FIPA [16] to communicate. ACL messages are objects and require protocols to allow their transport, which is provided by RMI (*Remote Method Invocation*). This specification is very useful in the event that the applications are executed from mobile devices, which have limited processing capabilities.

✓ **Agent Platform**

The set of roles that comprise the agent platform in PAINALLI can control and manage all of the architecture´s functionalities: applications, services, communication, output, reasoning capability and decision-making. Any modifications on agent behavior are carried out according to user preferences, knowledge obtained from previous interactions, and the available options for responding to a particular situation.

Figure 1 shows the elements that comprise PAINALLI.

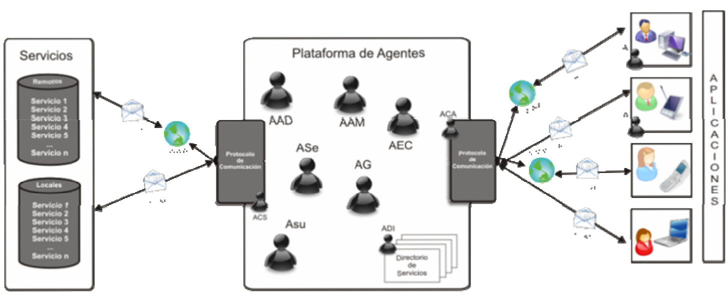


Fig. 1 PAINALLI architecture

In PAINALLI, the agents take on the following roles, which will subsequently help the external agents in the case study: **IA**: Interface agent, **ACA**: application communication agent, **CSA**: communication services agent, **DAA**: document administrator agent, **MAA**: message administrator agent, **DiA**: service directory manager agent, **SeA**; security manager agent, SuA: supervisor agent, **KEA**: knowledge extraction agent, **MA**: manager agent.

With both PAINALLI and web services, security has to control the users and the access given to each of the operations exposed through WSDL (*Web Services Description Language*) to all other users. It is also necessary for security to provide a set of bookstores that make it easy to work with security APIs for both the client and the server. When using SOAP protocol, one way of dealing with security is through WS-Security (*Web Service Security*), which allows the exchange of security Tokens between client and server (End-to-End security). This exchange of Tokens, as well as the mechanisms associated with encryption and signature, makes it possible to ensure the authentification, integrity and confidentiality of the operations that are carried out.

3 HC-MAS 1.0 System

The HC-MAS 1.0 system supports the administrative and medical communication process in a hospital. A hospital is an organization in which a great deal of information is generated and used, and where the key players in the process can be grouped into three categories: sources, recipients and intermediaries.

The information system in a healthcare environment, particularly that of a hospital, must be able to adopt the messages, formats, code and structure of medical histories in order to allow interoperability within the system.

The use of standards permits an increase in security, lowers costs and encourages market development. The implementation of standards and norms for all users, manufacturers and service providers, promotes the creation of more economical and stable solutions.

A document is one type of standard that is established by consensus, approved by a recognized entity, provides rules, guides or characteristics needed to carry out activities [19] . The use of standards makes it possible to achieve interoperability within the system and its components.

Interoperability is the ability of two or more systems or components to exchange information and use the information that has been exchanged. Syntactic interoperability (operative or functional) is the structure of communication, the equivalent of spelling and grammar rules. H7 (Health Level Seven) is one type of method that can be used to exchange messages or data within a health environment. Semantic interoperability contains the meaning of the communication, the equivalent of a dictionary or thesaurus. The CDA (Clinical Document Architecture) within HL7 is a structural and semantic. It is based on XML language, and SNOMED (Systematized Nomenclature of Medicine) terminologies and LOINC (Logical Observation Identifiers Names and Codes), both of which are examples of semantic standards.

For messaging and data exchange, HC-MAS 1.0 uses one of the HL7 standards, while for image exchange it uses DICOM. In order to best integrate the data and the HCE structure, the CDA standard proposed by HL7 is followed.

The following section presents the various roles that HC-MAS 1.0 needs to execute all the actions described in PAINALLI, in addition to the communication process for a hospital.

3.1 Agent Platform

After analyzing the communication process, the information, and user requests made by hospital personnel, it is possible to see how these elements can form an adaptive virtual organization whose administration depends on the variability of its products, users, etc. [20]. In order to completely satisfy the demands of the users in the communication process of a hospital, as well as those of the agent roles presented in PAINALLI, new roles can be added using the system´s open architecture. The new roles allow the system to conform to HCE by managing patient data, appointments, lab analyses, clinical tests, diet and patient care.

Figure 2 shows the PAINALLI architecture and the health agents that together manage and formalize the communication process in a hospital.

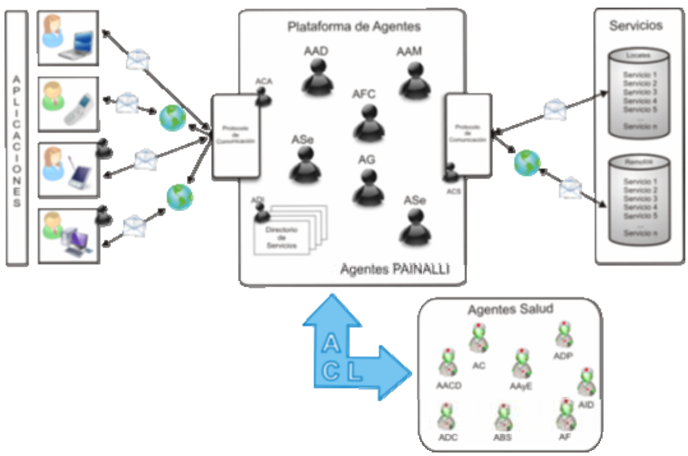


Fig. 2 Scenario of the communication process in a hospital

Health agents take on the following roles: **PDA (*Patient Data Agent*)**: manages the administrative and personal data of the patients; **SA (*Scheduling Agent*)**: manages appointments; **ALA (*Analysis and lab tests agent*)**: handles order forms for analyses and clinical studies; **IDA: (*Image diagnostic agent*)**: keeps track of messages for managing hemoderivative transfusion devices; **PA (*Pharmaceutical agent*)**: controls the templates for medical prescriptions; **CDAA (*Clinical document architecture agent*)**: assumes the same responsibility as the AG in PAINALLI, but with the roles corresponding to health agents.

4 Results and Conclusions

HC-MAS was installed in a hospital in Salamanca, Spain. Several interviews were carried out in early 2008. During the same period a number of hospital activities were timed. The goal of both activities was to have reference data to compare against the results obtained after installing the system.

Because the purpose of this paper is not to provide a full description of the various quality indicators, only a few descriptions will be identified. The results presented in this section are divided into 3 indicator groups. For each group there will be 2 variables described in detail, which makes it possible to gain an insight on: doctors, nurses, managers, patients and caregivers.

- **Assistance Indicators**

This information allows us to understand how much time doctors and nurses invest in writing reports and connected to the system.

Medical histories are the most important documents in a health system since they contain all of the information related to each of the patients. The indicator shows a 20% decrease in preparing these documents, which is due to the fact that HC-MAS can ¨write¨ the information directly in an electronic document. The indicator that most significantly demonstrates the effectiveness of using HC-MAS 1.0 is the interconnectivity that doctors can have with other doctors or clinical services in the hospital. What previously took on average 5 days to complete can now be done immediately.

- **Management Indicators**

Some of the different management indicators include: confirming patient eligibility and managing the costs involved in image diagnostics.

Confirming patient eligibility is an important step towards ensuring that services are rendered as quickly as possible, and that whoever receives treatment is eligible. A decrease of 32.44% in the waiting period involved in confirming eligibility makes it easier to grant services and more importantly ensures that 100% of the patients receiving services are in fact eligible. The indicator for managing the costs involved in image diagnostics is reduced significantly (90%) because with HC-MAS 1.0 it is no longer necessary to print images.

- **Patient and Caregiver Satisfaction Indicators**

Because the patients and healcare providers receive the services directly, their input carries more weight in the evaluation of system quality. Some of the indicators that demonstrate the level of satisfaction are: (i) the waiting time between appointments, which fell by almost 28% making patients feel that they are being better attended; (ii) the number of complaints, which fell from 84 to 35, resulting in 49 fewer complaints. To support this indicator, we analyzed the complaints. Of the 35 that were brought forth, 20 were related with variables from HC-MAS and 15 were related to aspects, such as administration and billing, that will be addressed in version 2.0.

The indicators reflect a significant decrease in time, which can now be spent on activities related to the professional and personal development of the employees.

One of the advantages of PAINALLI is the flexibility for adapting to different systems. HC-MAS 1.0 takes the agent platform architecture and adds new agents to solve the problem of communications in public or private healthcare environments.

Acknowledgements. This research has been partially supported by the project PET2008_0036.

References

1. Garcia Echevarría, S., Val Núñez, M.T.: New perspective in business organizations: lean Management. In: IDOE, Alcalá de Henares (1995)
2. Corchado, J.M., Tapia, D.I., Bajo, J.: A Multi-Agent Architecture for Distributed Services and Applications. Computational Intelligence (2009) (in press)
3. Jennings, N.R., Sycara, K., Woldridge, M.: A roadmap of agent research and development. In: Autonomous Agents and Multiagente Systems (1998)
4. Iglesias, C., Garijo, M., González, J.: A survey of agent oriented methodologies. In: Rao, A.S., Singh, M.P., Müller, J.P. (eds.) ATAL 1998. LNCS (LNAI), vol. 1555, pp. 317–330. Springer, Heidelberg (1999)
5. Bussmann, S.: Agent-oriented programming of manufacturing control tasks. In: Proc. 3rd Int. Conference on Multi-Agent Systems, ICMAS 1998 (1998)
6. Ferber, J., Gutknecht, O., Michel, F.: From agents to organizations: An organizational view of multi-agent systems. In: Giorgini, P., Müller, J.P., Odell, J.J. (eds.) AOSE 2003. LNCS, vol. 2935, pp. 214–230. Springer, Heidelberg (2004)
7. Castro, J., Kolp, M., Mylopoulos, J.: A requirements-driven software development methodology. In: Dittrich, K.R., Geppert, A., Norrie, M.C. (eds.) CAiSE 2001. LNCS, vol. 2068, p. 108. Springer, Heidelberg (2001)
8. Gateau, B., Boissier, O., Khadraoui, D., Dubois, E.: MOISE-Inst: An Organizational model for specifying rights and duties of autonomous agents. In: der Torre, L.V., Boella, G. (eds.) 1st International Workshop on Coordination and Organisation (2005)
9. Vazquez-Salceda, J., Dignum, V., Dignum, F.: Organizing Multiagent systems. Technical Report UU-CS-2004-015, Institute of Information and Computing Sciences. Utrecht University (2004)
10. Esteva, M., Rodriguez, J., Sierra, C., García, P., Arcos, J.: On the formal Specification of Electronic Institutions. In: Sierra, C., Dignum, F.P.M. (eds.) AgentLink 2000. LNCS (LNAI), vol. 1991, p. 126. Springer, Heidelberg (2001)
11. Argente, E., Botti, V., Carrascosa, C., Giret, A., Julian, V., Rebollo, M.: An Abstract Architecture for Virtual Organizations: The THOMAS project DSIC-II/13/08 (2008)
12. López Lita, R.: Comunicación: la clave del bienestar social. El Drac. (2000)
13. Cerami, E.: Web Services Essentials Distributed Applications with XML-RPC, SOAP, UDDI & WSDL, 1st edn. O'Reilly & Associates, Inc., Sebastopol (2002)
14. FIPA: Foundation for Intelligent Physical Agents (2005), `http://www.fipa.org` (retrieved 15 02, 2008)
15. Tapia, D.I., Bajo, J., Corchado, J.M., Rodríguez, S., De Paz, J.F., Pinzón, C., Cano, R.: Applying a Service Oriented Approach for Developing a Distributed Multi-Agent System for Health Care. International Journal of Computer Applications in Technology, IJCAT (2009) (in press)
16. van Bemmel, J.H., Musen, M.A.: Handbook of medical Informatics. Springer, Heidelberg (1997)
17. Camacho, D., Borrajo, D., Molina, J.M.: Intelligence Travel Planning: a 8ultiagente planning system to solve web problems in the e-tourism domain. International Journal on Autonomous Agents and Multiagent systems (2001)

Multi-agent System (MAS) Applications in Ambient Intelligence (AmI) Environments

Nayat Sánchez-Pi, Eleni Mangina, Javier Carbó, and José Manuel Molina

Abstract. Research in context-aware systems has been moving towards reusable and adaptable architectures for managing more advanced human-computer interfaces. Ambient. Intelligence (AmI) investigates computer-based services, which are ubiquitous and based on a variety of objects and devices. Their intelligent and intuitive interfaces act as mediators through which people can interact with the ambient environment. In this paper we present an agent-based architecture which supports the execution of agents in AmI environments. Two case studies are also presented, an airport information system and a railway information system, which uses spoken conversational agents to respond to the user's requests using the contextual information that includes the location information of the user.

Keywords: Multi-agent systems, Services Oriented Architectures, Mobile Context-Aware Systems, Multi-Agent Systems.

1 Introduction

Mobile technology is increasingly entering in all aspects of our life and in all sectors, opening a world of unprecedented scenarios where people interact with electronic devices embedded in environments that are sensitive to the presence of users. These context-aware environments combine ubiquitous information, communication, with enhanced personalization, natural interaction and intelligence.

Nayat Sánchez-Pi · Javier Carbó · José Manuel Molina
Computer Science Department
Carlos III University of Madrid
Avda de la Universidad Carlos III, 22. 28270. Colmenarejo, Spain
e-mail: {nayat.sanchez,javier.carbo,josemanuel.molina}@uc3m.es

Eleni Mangina
Computer Science Department
University College of Dublin
Dublin, Ireland
e-mail: {eleni.mangina}@ucd.ie

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 493–500.
springerlink.com

The use of this context offers the possibility to tailor a new type of advanced applications. The design and development of effective applications should definitely take into account the characteristics of the context from which a service is requested [9]. Contextual information can be either the type of the device exploited to access a service, or the location of the user, or its personal preferences, etc.

Although there is not a complete agreement on the definition of context, the most widely accepted is the one proposed by [3]: “Any information that can be used to characterize the situation of an entity is relevant to the interaction between a user and an application, including the user and the application themselves”. As can be observed from this definition, any information source can be considered context as long as it provides knowledge relevant to handle the communication between the user and the system. In addition, the user is also considered to be part of the contextual information. Kang et al [15] differentiate two types of context: internal and external. The former describes the user state (e.g. communication context and emotional state), whereas the latter refers to the environment state (e.g. location and temporal context). Most studies in the literature focus on the external context. However, although external information, such as location, can be a good indicator of the user intentions in some domains, in many applications it is necessary to take into account more complex information sources about the user state, such as emotional status [1] or social information [17]. External and internal context are intimately related, as can be seen in representative examples like service context and proactive systems.

Context information can be gathered from a wide variety of sources, which produces heterogeneity in terms of quality and persistence. As described in [11], static context deals with invariant features, whereas dynamic context is able to cope with information that changes. The frequency of such changes is very variable and can deeply influence the way in which context is obtained. It is reasonable to obtain largely static context directly from users, and frequently changing context from indirect means such as sensors.

Ambient Intelligence is associated to a society based on unobtrusive, often invisible interactions amongst people and computer-based services taking place in a global computing environment. A good point seen on the AmI vision is that the electronic or digital part of the ambience (devices) will often need to act intelligently on behalf of people [21]. In [22], O'Hare et al. advocate the use of agents as a key enabler in the delivery of ambient intelligence and ubiquitous environments. The components of ambience will need to be both reactive and proactive, behaving as if they were agents that act on behalf of people. If we assume that agents are abstractions for the interaction within an ambient intelligent environment, one aspect that we need to ensure is that their behavior is regulated and coordinated. For this purpose, we need rules that take into consideration the context (location, user profile, type of device, etc…) in which these interactions take place. Taking care this, the system needs an organization similar to the one envisaged by artificial agent societies. The society is there not only to regulate behavior but also to distribute responsibility amongst the member agents.

This paper is structured as follows: Section 2 describes the main characteristics of our agent-based architecture for providing context-aware adaptable systems and the new characteristics implemented; Section 3 presents the case of use in the domain of an airport; Section 4 presents another case of use with an intelligent speech-based interface in a railway domain and finally Section 5 present our conclusions and future work.

2 MAS Architecture for AmI

Current research has focused on agent and multi-agent implementations which support the ubiquitous provisioning of context-aware services to mobile devices. CONSORT [16] is a multi-agent architecture including a middle agent that translate sensor based raw representation of the locations into a conceptual one using a location ontology in two application domains that model the geographical space in a cognitive way: an intelligent assistant at a museum and a wireless-LAN based location system. Another is SMAUG [19] which is a multi-agent context-aware system that allows tutors and pupils of a university to fully manage their activities. SMAUG offers its users with context-aware information from their environment and also gives them a location service to physically locate every user of the system. Another one is IUMELA [18], is an intelligent modular-education learning assistant designed using a multi-agent systems in order to assist students in their career decision-making process. All the above research projects although successful to their own problem domain, do not provide a general architecture that can be applied to different scenarios and case studies. Within this paper our aim is to present the redesign our initial multi-agent system for AmI [20; 21], utilized in two different case studies. The redesign includes new features to support an intelligent speech-based interface.

Figure 1 shows the architecture of our previous work, while Figure 2 denotes our new proposal that allow us distribute contextual information between different agents.

New agent's functionalities state as follow:

Positioning agent main tasks rely on the user identification and user location into the environment. In order to do this, this agent needs to connect to Aruba Positioning System [20], to read the positioning information. This information consists of an (x,y) coordinates and the building and floor information of the system. Facilitator agent is the responsible of the services management and the discovering of services agent identification. Therefore, it carries out the matching between the user profile stored in the user agent and the different services, since they can reach an agreement with clients and communicate the most suitable services to them according clients preferences and profile. User agents main goals includes negotiation with the facilitator agent, recommend services to other user agents, trust in other agents, and manage and improve their internal profile to receive better services according to it. The information defined in the user profile stored in the user agent can be classified into two different groups:

-Personal information: user's name, gender, age, current language, skill level when interacting with dialog systems, pathologies or speech disorders;

-User preferences: This set is split into different statistic groups, which are defined according to a specific domain-knowledge. This new subdivision allows the

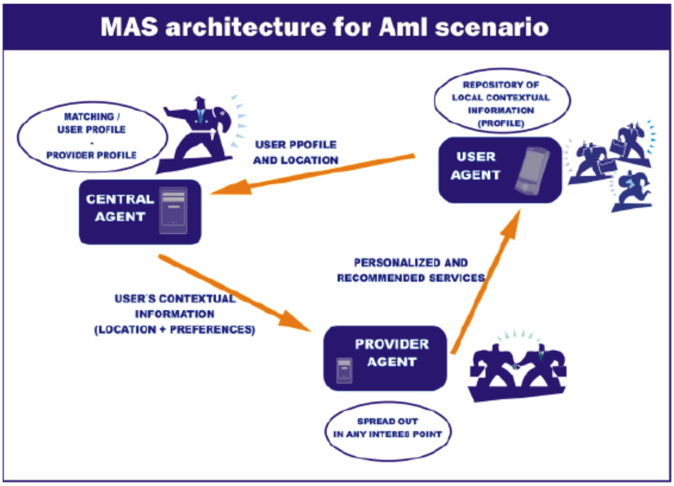


Fig. 1 MAS architecture (extracted from V. Fuentes, N. Sanchez-Pi, J. Carbó, J.M. Molina. Designing a Distributed Context–Aware Multi–Agent System" from Agent-Based Ubiquitous Computing, chapter of Agent-based Ubiquitous Computing, Series: Atlantis Ambient and Pervasive Intelligence - Vol. 1. E. Mangina, J. Carbó, J.M.Molina (Eds.). Ed. World Scientific. ISBN 978-90-78677-10-9).

system to infer the preferences of each user regarding specific queries included in the task (i.e. in a railway information system to know that a specific user usually requires timetables information) or relative to specific values of attributes to be used in these queries (i.e. the preferences for travelling during a specific part of the day or by using a specific type of train).

Although there are a high variety of applications in which conversational agents can be used. One of the most wide-spread is information retrieval that was the main reason to include conversational agents in this proposal.

To successfully manage the interaction with the users, conversational agents usually carry out five main tasks: automatic speech recognition (ASR), natural language understanding (NLU), dialog management (DM), natural language generation (NLG) and text-to-speech synthesis (TTS). These tasks are usually implemented in different modules.

According to the implementation and although other alternatives exist, JADELEAP agent platform was chosen since it is a Java-based and FIPA compliant agent platform, where agents communicate by sending FIPA ACL messages over a TCP/IP connection between different runtime environments on local servers or running on wireless devices. One local server hosts the positioning agent that rules over all the sensors of the domain (these sensors are just simulated inside the implementation of the central agent) that provide location information of users. Another server hosts the facilitator agent that matches the user profile with the services while one or more local servers host conversational agents acting on behalf of the different services. Each portable client device runs a JADE-LEAP

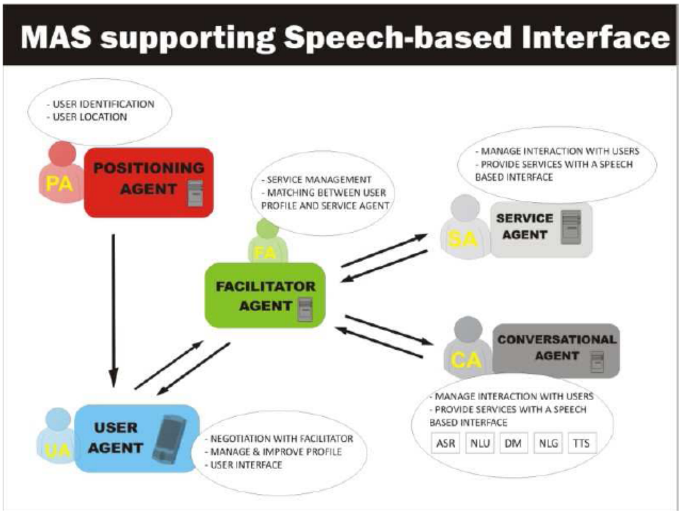


Fig. 2 MAS adequacy proposed new architecture to support speech-based interface

container that hosts one single agent that is used to provide a way to interact with the user (through a GUI), so the user can be informed and interact with the other agents running on different servers. JADE/LEAP platform logically connects the agents on the different servers with each other and with the corresponding user agents running on the users' devices.

3 Case Study I: Airport Scenario

The first case study of the proposed new architecture, presents the definition of an information system for an airport domain. What we precisely propose is a distributed approach based on agents to address the problem of using context-aware information to offer customized services to users in an airport domain, and especially for the Barajas-Madrid Airport and for services offered to clients, passengers, crew and staff of IBERIA airline. We are familiar to this problem since we previously developed a centralized system with the same final intention using Appear Networks Platform and Aruba Wi-Fi Location System [20; 21]. The current proposal includes agents implemented in JADE and LEAP that will make use of the user profiles to customize and recommend different services to other agents avoiding an obtrusive participation of a central server. First we assume an initial minimal profile known user agent: name, agent role, passport number, nationality and travel info (flight numbers, companies, origin and target). Figure 3 shows the console agents and the HTC touch terminal. We will describe how the system works by following all the messages exchanges when a passenger "Paul" arrives to the airport .When Paul enters into the system's wireless network. Aruba Positioning system discovers the Paul's agent position. Later, positioning agent provides Aruba positioning information to the Paul's user agent. Once Paul's user agent (running the LEAP code on the mobile device) knows its location sends it to the facilitator agent as well as the information regarding using a specific kind of service, in this case Paul's preferences include automated check-in service. After that, facilitator agent sends the automated check-in agent identification, the interaction between them begin and the provisioning of the service occur.

4 Case Study II: Railway Station Scenario

There is a high variety of applications in which conversational agents can be used. One of the most wide-spread is information retrieval. Some sample applications are tourist and travel information [8, 14], weather forecast over the phone [23], speech controlled telephone banking systems [12, 13], conference help [5, 10], etc. They have also been used for education and training, particularly in improving phonetic and linguistic skills: assistance and guidance to F18 aircraft personnel during maintenance tasks [4], dialog applications for computer-aided speech therapy with different language pathologies [2].

In this section we present a railway information system that includes a speech-based interface. We are familiar to this problem since we previously developed a centralized system with the same final intention using Appear Networks Platform and Aruba Wi-Fi Location System [6, 7]. To successfully manage the interaction with the users, conversational agents usually carry out five main tasks: automatic speech recognition (ASR), natural language understanding (NLU), dialog management (DM), natural language generation (NLG) and text-to-speech synthesis (TTS). These tasks are usually implemented in different modules.

The behaviour of the system is the following: In the first phase, the Aruba Positioning system discovers the user's position while he enters the Wi-Fi network in the Atocha Station. Later, positioning agent provides Aruba positioning information to the user agent. Once user agent knows its location sends it to the facilitator agent as well as the information regarding using a specific kind of service, in this case the user decides to ask for a conversational service. Following the set of phases, a conversational agent that provides the railway information and has been previously detected, asks the user agent about context information to be used during the interaction to provide the personalized service. Once this context information is received by the Context Manager included in the conversational agent, it loads the specific context profile characteristics. This information is then consulted by the rest of the modules in the conversational agent to personalize the provided service.

5 Conclusions

In this paper we presented middleware architecture for AmI environment to manage location-sensing systems and dynamically deploying services at suitable computing devices. Although it is a general framework in the sense that it is independent of any higher-level applications and location-sensing systems, we have designed and implemented a prototype system of the infrastructure and demonstrated its effectiveness in two practical applications: an airport and railway scenario. The main contributions are related to the adaptation of the MAS architecture. We have also presented two case studies: an airport information system and a railway information system, which uses spoken conversational agents to respond to the user's requests using the contextual information that includes the location information of the user.

Finally, we would like to point out further issues to be resolved. Since the framework presented in this paper has been applied to specific applications, we plan to evaluate it in terms of utility function of the MAS. The utility function will give a quantitative result of the effectiveness of the updated architecture compared to the initially implemented one. As a future work, we would like to evaluate our methodology in real environments.

Acknowledgements. This work has been partially supported by CICYT TIN2008-06742-C02-02/TSI, CICYT TEC2008-06732-C02-02/TEC, SINPROB, CAM MADRINET S-0505/TIC/0255 and DPS2008-07029-C02-02.

References

1. Callejas, Z., López-Cózar, R.: Influence of contextual information in emotion annotation for spoken dialogue systems. Speech Communication 50(5), 416–433 (2008)
2. Vaquero, C., Saz, O., Lleida, E., Marcos, J., Canalís, C.: VOCALIZA: An application for computer-aided speech therapy in Spanish language. In: Proc. of IV Jornadas en Tecnología del Habla, Zaragoza, Spain, pp. 321–326 (2006)
3. Dey, A., Abowd, G.: Towards a better understanding of context and context-awareness. In: Proc. of the 2000 Conference on Human Factors in Computer Systems (CHI 2000), New York, New Jersey, US, pp. 304–307 (2000)
4. Bohus, D., Rudnicky, A.: LARRI: A Language-Based Maintenance and Repair Assistant. In: Proc. of Multi-Modal Dialogue in Mobile Environments Conference (IDS 2002), Kloster Irsee, Germany, pp. 203–218 (2002)
5. Bohus, D., Grau, S., Huggins-Daines, D., Keri, V., Krishna, G., Kumar, R., Raux, A., Tomko, S.: Conquest - an Open-Source Dialog System for Conferences. In: Proc. of Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics, Rochester, NY, USA, pp. 9–12 (2007)
6. Griol, D., Sánchez-Pi, N., Carbó, J., Molina, J.M.: A Context-Aware Architecture to Provide Adaptive Services by means of Spoken Dialogue Interaction. In: Proc of ICAI 2009 – The 2009 International Conference on Artificial Intelligence, Las Vegas, USA (July 2009)
7. Griol, D., Sánchez-Pi, N., Carbó, J., Molina, J.M.: Context-Aware Approach for Orally Accessible Web Services. In: Proc. of IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technologies, Milán, Italy, September 2009, pp. 171–174 (2009)
8. den Os, E., Boves, L., Lamel, L., Baggia, P.: Overview of the ARISE project. In: Proc. of the European Conference on Speech Technology (Eurospeech 1999), Budapest, Hungary, pp. 1527–1530 (1999)

9. Fuentes, V., Pi, N.S., Carbó, J., Molina, J.M.: Reputation in user profiling for a context-aware multiagent system. In: Proc. of the 4th European Workshop on Multi-Agent Systems, EUMAS 2006, Lisbon, Portugal (2006)
10. Andeani, G., Fabbrizio, D.D., Gilbert, M., Gillick, D., Hakkani-Tur, D., Lemon, O.: Let's DISCOH: Collecting an Annotated Open Corpus with Dialogue Acts and Reward Signals for Natural Language Helpdesks. In: Proc. of IEEE 2006 Workshop on Spoken Language Technology (SLT 2006), Palm Beach, Aruba, pp. 218–221 (2006)
11. Henricksen, K., Indulska, J., Rakotonirainy, A.: Modeling context information in pervasive computing systems. In: Mattern, F., Naghshineh, M. (eds.) PERVASIVE 2002. LNCS, vol. 2414, pp. 167–180. Springer, Heidelberg (2002)
12. Hardy, H., Biermann, A., Inouye, R., McKenzie, A., Strzalkowski, T., Ursu, C., Webb, N., Wu, M.: The Amitiés system: Data-driven techniques for automated dialogue. Speech Communication 48, 354–373 (2006)
13. Melin, H., Sandell, A., Ihse, M.: Ctt-bank: A speech controlled telephone banking system - an initial evaluation. In: TMH-QPSR, vol. 1, pp. 1–27 (2001)
14. Glass, J., Flammia, G., Goodine, D., Phillips, M., Polifroni, J., Sakai, S., Sene®, S., Zue, V.: Multilingual spoken-language understanding in the MIT Voyager system. Speech Communication 17, 1–18 (1995)
15. Kang, H., Suh, E., Yoo, K.: Packet-based context aware system to determine information system user's context. Expert Systems with Applications 35, 286–300 (2008)
16. Kurumatani, K., Sashima, A., Izumi, N.: Location-mediated web services coordination in ubiquitous computing. In: IEEE Int. Conf. on Web Services (ICWS 2004), pp. 822–823 (2004)
17. Markopoulos, P., de Ruyter, B., Privender, S., van Breemen, A.: Case study: bringing social intelligence into home dialogue systems. Interactions 12(4), 37–44 (2005)
18. McGovern, E., Roche, B., Collier, R., Mangina, E.: IUMELA: A Lightweight Multi-Agent Systems Based Mobile Learning Assistant Using the ABITS Messaging Service. In: Indulska, J., Ma, J., Yang, L.T., Ungerer, T., Cao, J. (eds.) UIC 2007. LNCS, vol. 4611, pp. 1056–1065. Springer, Heidelberg (2007)
19. Nieto-Carvajal, I., Botía, J.A., Ruiz, P.M., Gómez-Skarmeta, A.F.: Implementation and Evaluation of a Location-Aware Wireless Multi-agent System. In: Yang, L.T., Guo, M., Gao, G.R., Jha, N.K. (eds.) EUC 2004. LNCS, vol. 3207, pp. 528–537. Springer, Heidelberg (2004)
20. Sánchez-Pi, N., Carbó, J., Molina, J.: Analysis and Design of a Multi-Agent System Using Gaia Methodology in an Airport Case of Use. In: Proc. of IBERAGENTS 2008. 7th Ibero-American Workshop in Multi-Agent Systems. In Conjunction with IBERAMIA 2008, Lisbon, Portugal (October 2008)
21. Sánchez-Pi, N., Carbó, J., Molina, J.: JADE/LEAP agents in an aml domain. In: Corchado, E., Abraham, A., Pedrycz, W. (eds.) HAIS 2008. LNCS (LNAI), vol. 5271, pp. 62–69. Springer, Heidelberg (2008)
22. O'Hare, G.M.P., O'Grady, M.J., Kegan, S., O'Kane, D., Tynan, R., Marsh, D.: Inteligent Agile Agents: Active Enablers for Ambient Intelligence. In: ACM's Special Interest Group on Computer-Human Interaction (SIGCHI), Ambient Intelligence for Scientific Discovery (AISD) Workshop, Vienna, April 25 (2004)
23. Zue, V., Sener, S., Glass, J., Polifroni, J., Pao, C., Hazen, T., Hetherington, L.: JUPITER: A telephone-based conversational interface for weather information. IEEE Transactions on Speech and Audio Processing 8, 85–96 (2000)

Agents to Help Context-Aware System in Home Care

Juan A. Fraile, Dante I. Tapia, Jesús A. Román, and Oscar García

Abstract. Context-aware systems capture information from the attributes located within their surroundings and deal with different ways of interacting with the user in its environment. This paper proposes a multi-agent system that processes and reasons the data it receives in order to identify and maintain a permanent fix on the location of a patient at home, while reliably and securely managing the infrastructure of services in its environment. The proposed multi-agent system was used to develop a prototype for controlling dependent persons in their home. The results obtained are presented in this paper.

Keywords: Context-Aware Computing, Home Care, Multi-agent system.

1 Introduction

The search for software environments that can adapt better and better to the demands of users and their environments leads us to context-aware systems. These systems store and analyze all the relevant information that surrounds us and constitutes a user environment. One of the environments in which the aforementioned system can be most useful is Home Care. The sharp increase in the number of dependent persons and the advanced state of technological development create the need to generate new solutions for Home Care environments [15] [9] – advanced applications that can be installed in the homes of dependent persons in order to improve their quality of life. Home Care requires the use of sensors and intelligent devices to build a distributed [3] environment in which household functions are

Juan A. Fraile
Pontifical University of Salamanca, c/ Compañía 5, 37002 Salamanca, Spain
e-mail: jafraileni@upsa.es

Dante I. Tapia · Jesús A. Román
Departamento de Informática y Automática, University of Salamanca,
Plaza de la Merced s/n, 37008 Salamanca, Spain
e-mail: {dantetapia,zjarg}@usal.es

Oscar García
School of Telecommunications, University of Valladolid, Spain
e-mail: ogargar@ulises.tel.uva.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 501–508.
springerlink.com

automated. In this respect, multi-agent systems [2] can facilitate the development of home care environments. Multi-agent systems have been studied recently as monitoring systems in the medical care [1] of sick patients and those suffering from Alzheimer's [8]. These systems can provide continuous support in the daily lives of these individuals [7], by predicting potentially dangerous situations, and handling physical and cognitive support for the person receiving assistance [4].

This article presents the Home Care Context-Aware Computing (HCCAC) multi-agent architecture, which is capable of supervising and monitoring persons in specific contexts. The objective of HCCAC is to facilitate the assistance of dependent users in their home. HCCAC integrates CBR-BDI [5] agents that are capable of learning beyond their initial knowledge, interacting autonomously with their environment and the system users, and adapting to the needs of the environment. Additionally, the HCCAC system can incorporate Case Based Planning (CBP) [16] mechanisms that provide the agents with a great deal of learning and adaptive abilities in a context-aware environment. The simple integration and interaction between intelligent agents, sensors and devices is what led us to propose the HCCAC architecture.

The remainder of the paper is structured as follows: section 2 describes the proposed system, the solutions offered by the Interpreter Agent, and a detailed description of the Interpreter Agent in HCCAC. Section 3 presents a case study in which the HCCAC was applied. Finally, section 5 presents the results and conclusions obtained from implementing a prototype Home Care scenario.

2 The Interpreter Agent in HCCAC

HCCAC [10] is a distributed system composed of intelligent agents that reason and interact with automatic control systems and autonomous components. The HCCAC architecture focuses primarily on tracking, control and notification. It is defined by the need to control distributed devices and to gather user information from Context-Aware environments [12] in a non-invasive and automatic way. In order to obtain and process context data and offer solutions to the user, the HCCAC system is based on a multi-agent architecture that is comprised of various types of intelligent agents [10]. The primary agent in HCCAC is the Interpreter Agent, which is integrated into the system. The purpose of this agent is to improve the quality of life for the user by providing efficient and relevant solutions in execution time. The most important characteristics of the system are: (i) the Interpreter Agent has reasoning capability; it can analyze and reason the context data gathered by the system and provide proactive solutions, (ii) the Interpreter Agent can easily adapt to the context within which it acts and (iii) gather sensor data and messages from other agents in order to provide efficient solutions. In order to meet the user´s expectations, the Interpreter Agent is based on Case Based Reasoning (CBR).

Case Based Reasoning uses past experiences to solve problems [5]. The objective of CBR systems it to solve new problems by adapting solutions that have been used to solve similar problems in the past. The Interpreter Agent also utilizes the concept of Case Based Planning (CBP) to generate solution plans, using past experiences and planning strategies. One example of how the Interpreter Agent works is its ability to apply the user´s temperature preferences when the user´s presence is detected within the environment. The system stores user preferences for the desired temperature. The system also stores the case base for the same user with past temperature preferences and similar temperature conditions, including external temperatures. The Interpreter Agent uses this information and similar case base plans to automatically adjust the control mechanism for the temperature when it detects the presence of the user. The system can thus maintain the desired temperature for each user within a specific context. The following section describes the design of the Interpreter Agent in greater detail.

2.1 Interpreter Agent Design

Agents that are designed and implemented by CBR systems can reason autonomously and adapt to changes in their environment [9]. These abilities satisfy two of the most important characteristics for the Interpreter Agent as mentioned in the previous point: (i) reasoning ability and (ii) facility in adapting to its surroundings. The other important characteristic is the ability to use sensors and other agents to gather information from its environment. Therefore, in order to design and implement the Interpreter Agent, it is necessary to consider the exchanges of information among the system agents. The FIPA[1] specification can be considered as a valid standard for the communication between agents. To design the Interpreter Agent, the Agent Unified Modeling Language (AUML[2]) methodology was used since it provides the mechanisms required to obtain a design that is detailed enough to simplify the implementation phase.

The Belief Desire Intention (BDI) [11] model is a solid foundation for modeling and applying the internal behavior of the agents. The BDI model allows us to perceive the agent as an entity that is searching for an objective and that behaves rationally. The CBR system and BDI agents can be connected if cases are implemented as beliefs, intentions and desires that lead to the resolution of a problem. For a smooth transition between the design phase and the implementation phase, the CBR-BDI paradigm must be supported in the implementation phase.

The (JADE) JavaAgent Development Framework [19] is a good option for developing agent-based applications. Jadex (JADE eXtension) [17] is the implementation of a hybrid agent (reactive and deliberative) architecture for representing the state of JADE agents that follow the BDI model. Jadex is designed to be easily integrated into JADE by simply adding a packet. The primary objective is to facilitate the use of reasoning concepts during the implementation.

As seen from the outside, the Interpreter Agent is a black box that receives and sends messages. The following section will attempt to explain the functioning of the Interpreter Agent in greater detail by using the figure below. The Interpreter Agent, as previously described, will be implemented as a Jadex agent. To do so, some variations in the Jadex architecture are introduced, as shown in figure 1.

---

[1] www.fipa.org
[2] www.auml.org

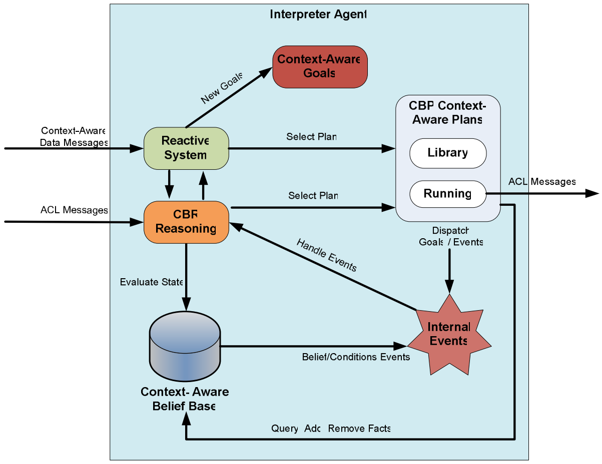


Fig. 1 Overview of the Interpreter Agent Architecture

Figure 1 provides a summary of the architecture for the Interpreter Agent. All messages received by the Interpreter Agent, as well as internal events and new goals, are the first steps towards the internal reactions and the deliberative and reasoning mechanisms processed by the Interpreter Agent. The most significant new feature of the Interpreter Agent´s design, as seen in figure 1, is the integration of a CBR reasoning engine and a reactive system that gathers data from the sensors and control systems. This makes the design unique in its conception and reasoning capabilities. Based on the results from the CBR reasoning engine, the Interpreter Agent sends plans through the CBP Context-Aware Plans. These plans can be executed immediately as events, or new plans can be generated and stored in the context-aware library to be executed at a future time. The execution of plans can modify the base of context-aware beliefs, send messages to other agents, create new context-aware goals or produce internal events. These action plans can respond to sensors installed in the system and facilitate the user´s daily tasks, making the time spent by the user in the context-aware environment much more comfortable. The plans are used by different types of HCCAC system agents that manage the active devices. The functionality implemented in Java classes can also be incorporated in other similar systems. The next section presents a low level AUML design for the Interpreter Agent, followed by the implementation of Jadex.

As shown in figure 2, the AUML design produces a diagram of capabilities and services for the Interpreter Agent, the most important agent within the HCCAC architecture. The agent has five capabilities and four services, as described in figure 2. The capabilities are: (i) P-Solution, (ii) C-Sensor, (iii) S-Plans, (iv) St-Data and (v) E-Result. The services are: (i) Provide Information (ii) Describe Plan, (iii) Provide Plan Result and (iv) Component Task Assignment.

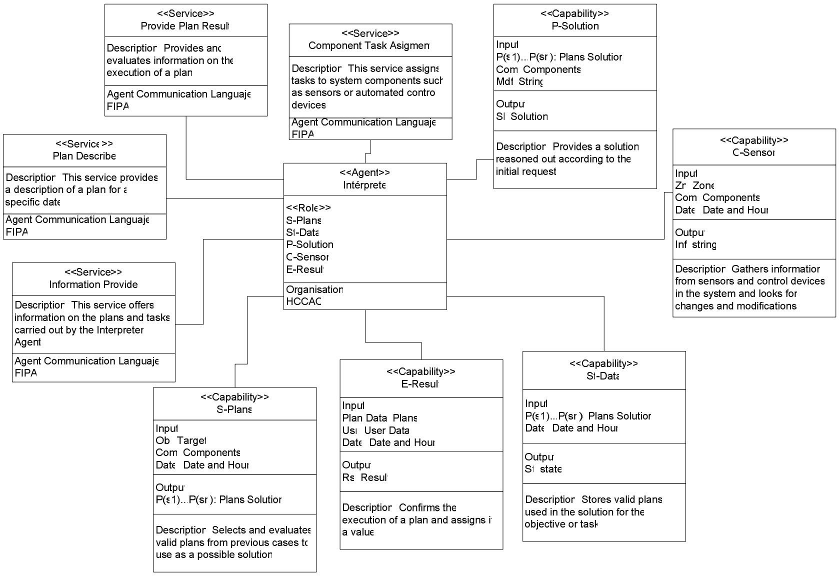


Fig. 2 Diagram of the different capabilities and services for the Interpreter Agent

3 Applying HCCAC in Context-Aware Environments

The case study presented in this paper uses the HCCAC system to develop a prototype for improving a patient´s quality of life at home. The system gathers information from the sensors that use automatic control systems to capture data and interact with the environment. The sensors installed in the environment primarily pick up location-aware information on the user. The system also gathers information related to the temperature inside the patient´s home, smoke detectors, flooding and lighting conditions in the areas most frequented by the user.

Figure 3 illustrates how the information provider agents are directly connected to the devices capturing the information. The application in this case is comprised of six modules: (i) to control the patient´s location (ii) to control the lighting conditions in the home (iii) to control temperature (iv) to control gas escapes (v) to control flooding and (vi) to control smoke detection. The Identification Resources Agent is responsible for identifying and either accepting or rejecting the data submitted by the information providers. Its task is to oversee the information provider agents that are incorporated into the system. All of the data are stored in the system and interpreted by the Interpreter Agent, which is also responsible for processing the stored information, obtaining, for example, the desired temperature and lighting, and then using the context-aware application to apply the data to the automated control devices. With the help of the automated control devices, the Interpreter Agent can also control the detection of flooding, smoke, and gas escapes in the patient´s surroundings.

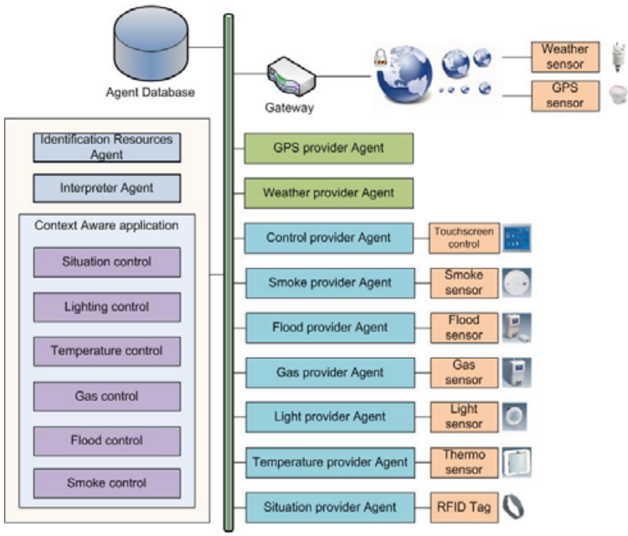


Fig. 3 Home Care context-aware application

Throughout the sequence of operations carried out by the HCCAC agents, it is important to consider the transfer of information that takes places in the system. Low level data are gathered from the patient´s environment and then stored in the information system as high level data. As a result the data can be interpreted more quickly and used more easily. The information provider agents and the Interpreter Agent are responsible for carrying out this task together. The case study presented in this paper explains how the agents defined in the system interact with the sensors installed in the environment, and how they influence the automated control devices and the patient.

Additionally, the patients can interact with their surroundings at any given time, establishing the parameters that control the function of each application. Just as provider agents gather context information, they can also receive orders to execute events from devices or automated control device systems. In this way, for example, the administrator can determine which patients can be monitored by the system and can also allow or deny patients access to certain areas in the home.

4 Results and Conclusions

HCCAC acts like a global system in context-aware systems. It not only captures information from its surroundings and responds users requests, but it also collaborates with the information system to constantly evaluate the attributes of the user´s context and provide proactive solutions. The solutions provided by HCCAC are supported by a vast knowledge base that the system continually stores and processes. HCCAC is a novel system compared to others context-aware types of works [6] [13] [14] [18]. Other studies have focused on gathering data from the user´s position. Others, such as [16], in addition to locating users within the context, try to improve the communication in a hospital center between patients and medical personnel by capturing such context attributes as weather, the patient´s state or the user´s role. However, the new services offered by HCCAC allow a closer and more natural and indirect interaction. The user can perform daily tasks and receive support from an intelligent context without needing direct interaction. As a result, the user does not actually need to learn to use the system since the HCCAC system itself manages the environment, and user satisfaction is notably increased.

Although there is still much work to be done, the system prototype developed in this study improves security in the homes of dependent persons through the use of control and alert devices. It also provides additional services that automatically react to emergency situations. As a result, HCCAC creates a context-aware environment that facilitates the development of distributed intelligent systems and provides services to dependent persons at home. This makes it possible to automate certain monitoring tasks and improve the quality of life for those individuals. Furthermore, the proper use of mobile devices facilitates social interactions and knowledge transfer. Our future work will focus on obtaining a model that can define contexts and improving the proposed prototype and testing it on patients with different needs and characteristics.

Acknowledgements. This work was supported in part by the projects JCYL SA071A08 and by the Ministerio de Ciencia e Innovación project PET2008_0036.

References

1. Angulo, C., Tellez, R.: Distributed Intelligence for smart home appliances. Tendencias de la minería de datos en España. Red Española de Minería de Datos, Barcelona, España (2004)
2. Ardissono, L., Petrone, G., Segnan, M.: A conversational approach to the interaction with Web Services. Computational Intelligence 20, 693–709 (2004)
3. Augusto, J.C., McCullagh, P.: Ambient Intelligence: Concepts and Applications. Invited Paper by the International Journal on Computer Science and Information Systems 4(1), 1–28 (2007)
4. Bahadori, S., Cesta, A., Grisetti, G., Iocchi, L., Leone1, R., Nardi, D., Oddi, A., Pecora, F., Rasconi, R.: RoboCare: Pervasive Intelligence for the Domestic Care of the Elderly. AI*IA Magazine Special Issue (2003)
5. Bajo, J., De Paz, J.F., Tapia, D.I., Corchado, J.M.: Distributed Prediction of Carbon Dioxide Exchange Using CBR-BDI Agents. International Journal of Computer Science (INFOCOMP), 16–25 (February 2007); ISSN 1807-4545, Special Edition
6. Burrell, J., Gay, G.: E-graffiti: evaluating real-world use of a context-aware system. Interacting with Computers – Special Issue on Universal Usability 14(4), 301–312 (2002)
7. Corchado, J.M., Bajo, J., de Paz, Y., Tapia, D.: Intelligent Environment for Monitoring Alzheimer Patients, Agent Technology for Health Care. Decision Support Systems 34(2), 382–396 (2008)

8. Corchado, J.M., Bajo, J., Abraham, A.: GERAmI: Improving the delivery of health care. IEEE Intelligent Systems. Special Issue on Ambient Intelligence (March/April 2008)
9. Corchado, J.M., Pavón, J., Corchado, E., Castillo, L.F.: Development of CBR-BDI Agents: A Tourist Guide Application. In: Funk, P., González Calero, P.A. (eds.) ECCBR 2004. LNCS (LNAI), vol. 3155, pp. 547–559. Springer, Heidelberg (2004)
10. Fraile, J.A., Bajo, J., Corchado, J.M.: Applying Context-Aware Computing in Dependent Environments. In: Mira, J., Ferrández, J.M., Álvarez, J.R., de la Paz, F., Toledo, F.J. (eds.) IWINAC 2009. LNCS, vol. 5602, pp. 85–94. Springer, Heidelberg (2009)
11. Holvoet, T., Valckenaers, P.: Beliefs, desires and intentions through the environment. In: Proceedings of AAMAS 2006, pp. 1052–1054. ACM, New York (2006)
12. Jameson, A.: Modeling both the context and the user. Personal and Ubiquitous Computing 5(1), 29–33 (2001)
13. Kerer, C., Dustdar, S., Jazayeri, M., Gomes, D., Szego, A., Caja, J.A.B.: Presence-aware infrastructure using web services and RFID technologies. In: Proceedings of the 2nd European Workshop on Object Orientation and Web Services, Oslo, Norway (2004)
14. Khedr, M., Karmouch, A.: Negotiation Context Information in Context-Aware Systems. IEEE Intelligent Systems 19(6), 21–29 (2004)
15. Majeed, B.A., Brown, S.J.: Developing a well-being monitoring system - Modeling and data analysis techniques. Applied Soft Computing 6, 384–393 (2006)
16. Muñoz, M.A., Gonzalez, V.M., Rodríguez, M., Favela, J.: Supporting context-aware collaboration in a hospital: An ethnographic informed design. In: Favela, J., Decouchant, D. (eds.) CRIWG 2003. LNCS, vol. 2806, pp. 330–344. Springer, Heidelberg (2003)
17. Pokahr, A., Braubach, L., Lamersdorf, W.: Multi-Agent Programming. In: Jadex: A BDI Reasoning Engine. Kluwer Book, Dordrecht (2005)
18. Priyantha, N.B., Chakraborty, A., Balakrishnan, H.: The cricket location-support system. In: Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, pp. 32–43. ACM Press, New York (2000)
19. Zhao, Z., Belloum, A., de Laat, C., Adriaans, P., Hertzberger, B.: Using Jade agent framework to prototype an e-Science workflow bus. In: Seventh IEEE International Symposium on Cluster Computing and the Grid, CCGRID 2007, pp. 655–660 (2007)

SDL Ontology for Specifying Systems Based on Finite State Machines

Marina Bagić Babac and Marijan Kunštić

Abstract. Specification and Description Language (SDL) is an object-oriented and formal standardized language for the specification of complex, event-driven, real-time and interactive applications involving many concurrent activities that communicate using discrete signals. Using SDL formal model for system specification we bridge the gap between ideas in our minds and the actual implementation of the system. In this paper we propose the ontology for the basic SDL elements. We also propose a formal framework of SDL Markup Language as a medium for translating SDL model to SDL ontology.

1 Introduction

The motivation for this paper lies in the authors belief that a semantically well-defined formal model for the specification of concurrent real-time systems is a good starting point in the process of developing a communications software. We have chosen Specification and Description Language (SDL) for a few reasons. First, it is a standardized language by ITU-T as Z.100 recommendation and is defined for the specification and description of complex, event-driven, real-time and interactive applications involving many concurrent activities that communicate using discrete signals. Then, it has both graphical and textual notation enabling the non-experts in the engineering and software development to easily follow the workflow of a systems' processes.

Inspired with the idea of Semantic Web [2, 3, 4] and triggered by Petri net ontology development [5], we have come to the idea of SDL ontology development. As

Marina Bagić Babac · Marijan Kunštić
University of Zagreb, Faculty of Electrical Engineering and Computing, HR-10000 Unska 3, Zagreb, Croatia
e-mail: {marina.bagic,marijan.kunstic}@fer.hr

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 509–517.
springerlink.com

the Semantic Web is the Web of data that is understandable by computers, we have proposed SDL ontology concept, but also the idea how to use this ontology without advanced knowledge of either the Semantic Web or the other techniques besides SDL language. We have proposed a formal framework of SDL Markup Language (SDL-ML) as a medium for translating SDL model to SDL ontology.[1]

2 Specification and Description Language Modeling

SDL language supports two equivalent notations, graphical (SDL-GR) and textual (SDL-PR). The textual notation SDL-PR uses the textual syntax only. The graphical notation SDL-GR not only has graphical components, but also some textual parts that are identical with the textual representation SDL-PR. This is because some specifications, such as the specification of data and signals, are more naturally specified textually [1].

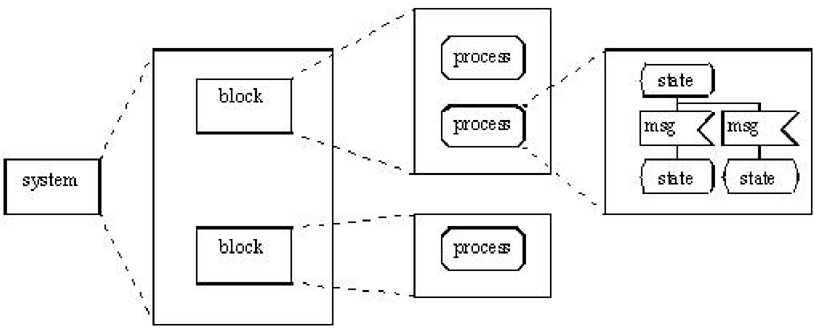


Fig. 1 SDL Architecture

The overall design (Figure 1) is called the system and everything that is outside the system is called the environment. There is no specific graphical representation for the system but the block representation can be used if needed. A block (or an agent) is an element in the system structure. There are two kinds of agents: blocks (meaning that a block can contain another block) and processes. A system is the outermost block. A block is a structuring element that does not imply any physical implementation on the target.

A process is a finite state machine (FSM) based task and has an implicit message queue to receive messages. It is possible to have several instances of the same process running independently. This is also one of the reasons for using SDL for describing FSM-based systems as it is built on it.

[1] This work was carried out within research project 036-0362027-1640 "Knowledge-based network and service management", supported by the Ministry of Science, Education and Sports of the Republic of Croatia.

3 SDL Ontology Core Development

In this section we develop the core SDL ontology, i.e. we put the basic concepts of SDL system modeling into the relations. The ontology can be expressed and depicted with various modeling techniques and different tools. However, the meaning of the relations among concepts should remain the same. The relations are what the Semantic Web is really about. As search engines today do not ”understand” the relations among different concepts, we need a different Web - a Semantic Web which will put each concept into the context and therefore provide us with better information once we need it.

For core SDL ontology we model the system, blocks and processes as well as the states and transitions (signals). The advanced features of SDL including timers, data structures, etc. can be derived once the basic concepts were agreed upon.

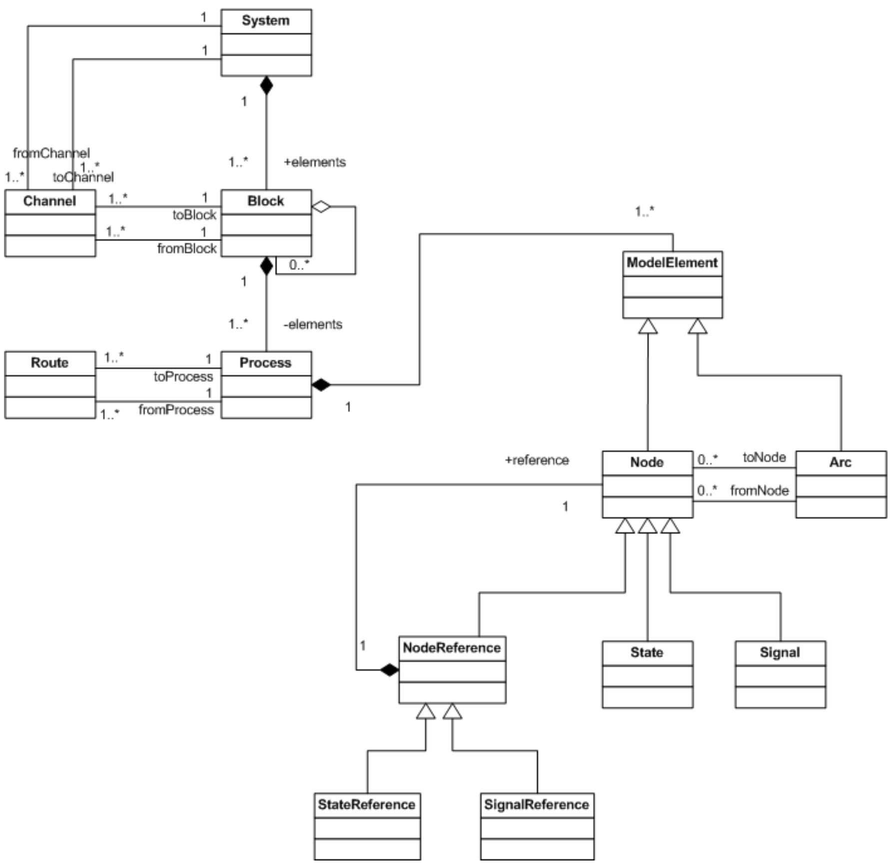


Fig. 2 SDL model ontology - UML hierarchy of core SDL concepts

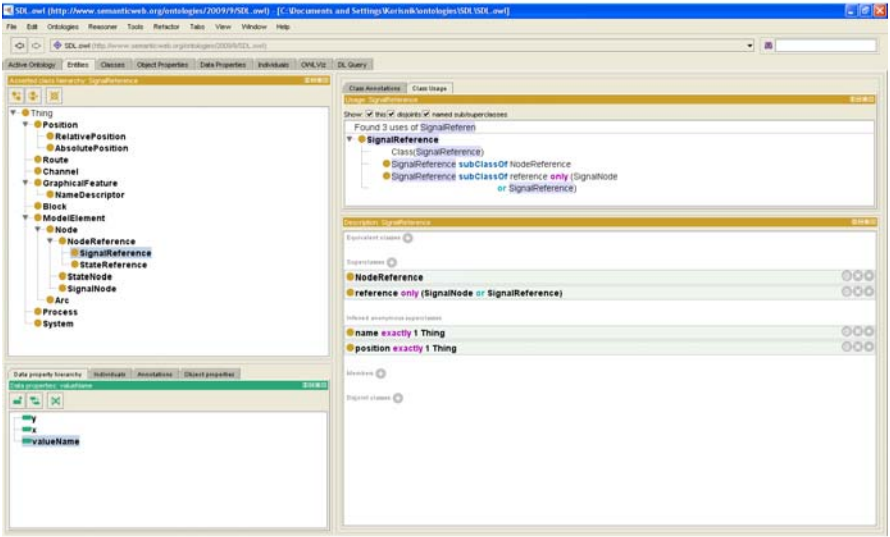


Fig. 3 SDL Ontology with Protégé Tool

We start with the UML representation of the SDL ontology (Figure 2) and then we translate it to OWL language using Protégé Tool [6]. In order to use UML notation we follow the generally adopted convention [8]. We could have translated SDL formal model directly to OWL, but we find UML suitable to express the whole SDL hierarchy model (both the system and process elements) in one figure. Besides, UML has a strong relations semantics which are suitable to unambiguously define the relations in SDL. For instance, UML composition is used to describe that a system contains blocks or agents, etc.

Relations in SDL model between the concepts of system, block, processes, etc. are translated to OWL properties, e.g. the concept of system is translated to OWL class System with the attribute *hasBlock* indicating that each instance of class System has at least one block. Similarly, we define the properties for Channel and Route entities, *fromChannel* and *toChannel*, *fromRoute* and *toRoute*, etc.

4 SDL Markup Language

As we have defined the SDL Ontology, we are triggered to impose it to the both SDL and non-SDL users. Our intention is to save their time and energy in the process of using it. Therefore, we believe that a step toward the automated translation from SDL specification to SDL Ontology should be developed. In this paper we outline this idea by introducing the fundamentals for SDL Markup Language (SDL-ML).

Table 1 Mapping SDL-ML to OWL

SDL-ML ELEMENTS	OWL ELEMENTS
System <sdl : system name=”name”> </sdl:system>	Class **System** $hasName \in [System, Literal]$ $hasSignal \in [System, Signal]$ $hasBlock \in [System, Block]$ $hasChannel \in [System, Channel]$
Block <sdl : block name=”name”> </sdl:block>	Class **Block** $hasName \in [Block, Literal]$ $hasSignal \in [Block, Signal]$ $hasBlock \in [Pro, Block]$ $hasRoute \in [Channel, Route]$
Process <sdl : process name=”name”> </sdl:process>	Class **Process** $hasName \in [Process, Literal]$ $hasSignal \in [Process, Signal]$ $hasState \in [Pro, Process]$
Channel <sdl : channel name=”name”> <from> </from> <to> </to> <with> </with> </sdl:channel>	Class **Channel** $hasName \in [Channel, Literal]$ $fromChannel \in [Channel, Block]$ $toChannel \in [Block, Channel]$ $withSignal \in [Channel, Signal]$
Route <sdl : route name=”name”> <from> </from> <to> </to> <with> </with> </sdl:route>	Class **Route** $hasName \in [Route, Literal]$ $fromRoute \in [Route, Process]$ $toRoute \in [Process, Route]$ $withSignal \in [Route, Signal]$
Signal <sdl : signal name=”name” type=”incoming/outgoing”> </sdl : signal>	Class **Signal** $hasName \in [Signal, Literal]$ $hasType \in [Signal, Literal]$
State <sdl:state name=”name”> <sdl:input> </sdl:input> <sdl:output> </sdl:output> <sdl:nextstate> </sdl:nextstate> </sdl:state>	Class **State** $hasName \in [Signal, Literal]$ $hasInputSignal \in [State, Signal]$ $hasOutputSignal \in [State, Signal]$ $hasNextState \in [State, State]$

SDL-ML is meant to be SDL interchange format that is independent of specific tools and platforms. Moreover, the interchange format needs to support extensions of SDL. Since this is our first proposal of this language, we shall leave the liberties to change or add some of the elements or their attributes in the future. However, this main set of elements should retain its semantics. On the other hand, the main problem for automating the process of conversion of SDL or SDL-ML to OWL lies in the hierarchy structure of both. In SDL (or SDL-ML) the hierarchy exists among structural elements i.e. a system contains blocks, a block contains another blocks or processes, etc. This is well explained by UML composition relation because the block does not exist without its system, or the process does not exist without its block and system. However, OWL hierarchy is given by inheritance concept, so the verbatim conversion is not possible. In the Table 1 we give an overview of these elements translation; OWL elements are classes with properties and their domain and range in angle brackets.

5 Case Study: FIPA Request Protocol for Multi-agent Systems

Here we give the simple example of FSM-based system. We have chosen a multi-agent system where each agent acts upon FSM and communicates with another agents in the system using specified protocols. We describe one of the standardized protocols, FIPA request protocol (Figure 4). The protocol is well explained with only two agents in the system, but the number of agent instances for the real system can be arbitrary;

The FIPA Request Interaction Protocol allows one agent to request another to perform some action. The Participant processes the request and makes a decision whether to accept or refuse the request. If an explicit agreement is required, then the Participant communicates an "agree". Once the request has been agreed upon, then the Participant sends a "failure" if it fails to fill the request, an "inform" if it successfully completes the request [7].

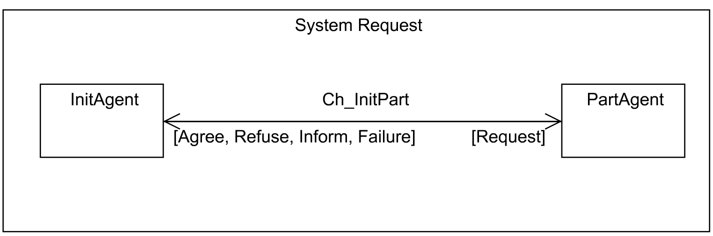


Fig. 4 SDL FIPA Request Protocol Specification

We describe the system with SDL blocks (Figure 4) where the descrete signals (messages) exchanged between agents are specified. Then, we specify the internal agent's behaviour with states and responses to these messages (Figure 5). SDL ontology for the system is built upon core SDL ontology (with Protégé Tool). We then create the .xml document in SDL-ML language for the system (Table 2).

Table 2 SDL-ML Code for FIPA Request Interaction Protocol

```
<sdl:system name=”RequestSystem”>
   <sdl:block name=”InitiatorAgent”>
      <sdl:process name=”InitiatorAgent”>
         <sdl:state name=”Idle”>
            <sdl:output>Request</sdl:output>
               <sdl:nextstate>wait</sdl:nextstate>
         < /sdl:state>
         <sdl:state name=”wait”>
            <sdl:input>Refuse</sdl:input>
               <sdl:nextstate>idle</sdl:nextstate>
            <sdl:input>Agree</sdl:input>
               <sdl:nextstate>waitAgree</sdl:nextstate>
         < /sdl:state>
         <sdl:state name=”waitAgree”>
            <sdl:input>Inform</sdl:input>
               <sdl:nextstate>idle</sdl:nextstate>
            <sdl:input>Failure</sdl:input>
               <sdl:nextstate>idle</sdl:nextstate>
         < /sdl:state>
      < /sdl:process>
   </sdl:block>
   <sdl:block name=”ParticipantAgent”>
      <sdl:process name=”ParticipantAgent”>
         <sdl:state name=”idle”>
            <sdl:input>Request</sdl:input>
               <sdl:output>Refuse</sdl:output>
                  <sdl:nextstate>idle</sdl:nextstate>
               <sdl:output>Agree</sdl:output>
                  <sdl:output>Inform</sdl:output>
                     <sdl:nextstate>idle</sdl:nextstate>
                  <sdl:output>Failure</sdl:output>
                     <sdl:nextstate>idle</sdl:nextstate>
      </sdl:process>
   </sdl:block>
   <sdl:channel name=”Ch-Init-Part”>
      <sdl:from>InitiatorAgent</sdl:from>
         <sdl:to>ParticipantAgent</sdl:to>
            <sdl:with>Request</sdl:with>
      <sdl:from>ParticipantAgent</sdl:from>
         <sdl:to>InitiatorAgent</sdl:to>
            <sdl:with>Agree</sdl:with>
            <sdl:with>Refuse</sdl:with>
            <sdl:with>Inform</sdl:with>
            <sdl:with>Failure</sdl:with>
   </sdl:channel>
</sdl:system>
```

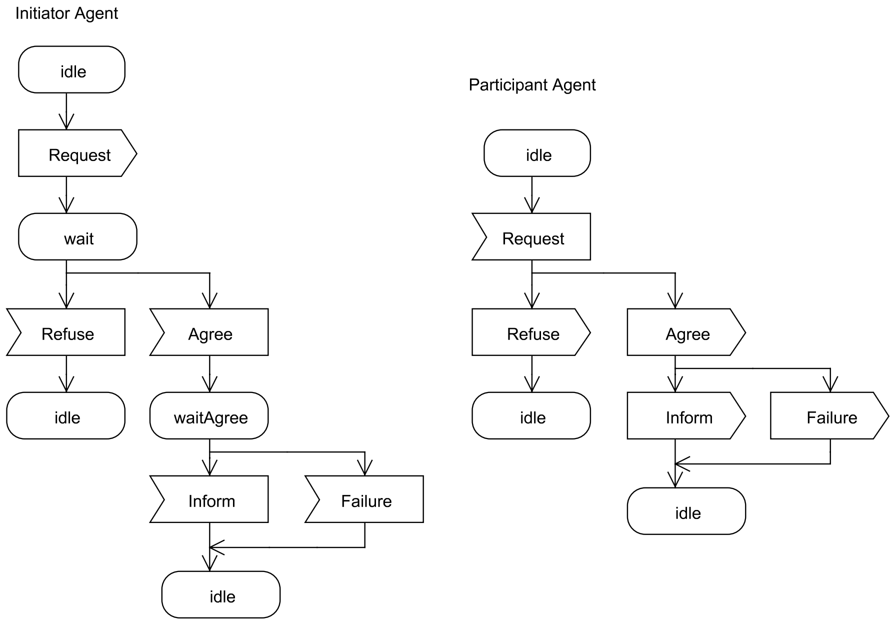


Fig. 5 Agents Communicating via FIPA Request Protocol

6 Conclusion

In this paper we have given our contribution to the efforts of Semantic Web development by introducing a new ontology for concurrent and FSM-based systems specification. Particularly, we have chosen Specification and Description Language, a ITU-T standard Z.100 to develop the ontology for SDL system and process specification. The ontology was developed using UML and OWL languages. We have also suggested the development of SDL Markup Language as a medium step towards SDL to OWL automated mapping.

References

1. Ellsberger, J., Hogrefe, D., Sarma, A.: SDL: Formal Object-Oriented Language for Communicating Systems. Prentice Hall PTR, Englewood Cliffs (1997)
2. Walton, C.: Agency and the Semantic Web. Oxford University Press, Oxford (2006)
3. Antoniou, G., Van Harmelen, F.: A semantic web primer, 2nd edn. MIT Press, Cambridge (2008)
4. Mika, P.: Social Networks and the Semantic Web. Semantic Web and Beyond, vol. 5 (2007) ISBN: 978-0-387-71000-6
5. Gašević, D., Devedžić, V.: Petri net Ontology. Knowledge-Based Systems 19(4), 220–234 (2006)

6. Horridge, M.: A Practical Guide To Building OWL Ontologies Using Protégé 4 and CO-ODE Tools, Edition 1.2, University of Manchester (2009)
7. FIPA: FIPA Request Interaction Protocol Specification, 2002/12/03 (2002)
8. Protégé Community of Practice, UMLBackendMapping (2009), http://protege.cim3.net/cgi-bin/wiki.pl?UMLBackendMapping

Applying Data Mining Techniques to Stock Market Analysis

Gabriel Fiol-Roig, Margaret Miró-Julià, and Andreu Pere Isern-Deyà

Abstract. The stock market can be viewed as a particular data mining and artificial intelligence problem. The movement in the stock exchange depends on capital gains and losses and most people consider the stock market erratic and unpredictable. However, patterns that allow the prediction of some movements can be found. Stock market analysis deals with the study of these patterns. It uses different techniques and strategies, mostly automatic that trigger buying and selling orders depending on different decision making algorithms. It can be considered as an intelligent treatment of past and present financial data in order to predict the stock market future behavior. Therefore it can be viewed as an artificial intelligence problem in the data mining field. This paper aims to study, construct and evaluate these investment strategies in order to predict future stock exchanges. Firstly, data mining techniques will be used to evaluate past stock prices and acquire useful knowledge through the calculation of some financial indicators. Next artificial intelligence strategies will be used to construct decision making trees.

Keywords: Data Mining, Artificial Intelligence, Stock Market, Investment Strategies, Decision Trees.

1 Introduction to the Problem

The problem considered deals with the design and validation of a data mining decision system which generates buying and selling orders in the stock market [1]. The decision system's design has three main phases as indicated in Figure 1: the data processing phase, the data mining phase and the evaluation phase.

The data processing phase selects from the raw data base a data set that focuses on a subset of attributes or variables on which knowledge discovery is to be

---

Gabriel Fiol-Roig · Margaret Miró-Julià · Andreu Pere Isern-Deyà
Departament de Ciències Matemàtiques i Informàtica
Universitat de les Illes Balears
Ctra. de Valldemossa km 7,5, 07122 Palma de Mallorca, Spain
e-mail: {biel.fiol,margaret.miro}@uib.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 519–527.
springerlink.com

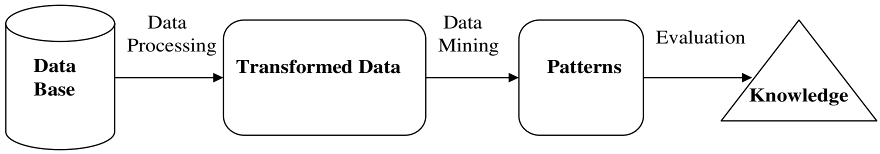


Fig. 1 The Decision System

performed. It also removes outliers and redundant information, and uses financial indicators to represent the processed data by means of an Object Attribute Table (OAT). The data mining phase converts the data contained in the OAT into useful patterns, in particular decision trees are found [2]. The evaluation phase proves the consistency of pattern by means of a testing set. The positively evaluated decision system can then be used in real world situations that will allow for its validation.

2 The Data Processing Phase

The original data base is formed by financial information available in different web pages. In particular, data from the American stock market, available from Google finance, have been used. Out of all the variables originally considered the target data is formed by: date (D), opening price (O), closing price (C), daily high (H), daily low (L) and daily volume (V) as illustrated in Table 1.

Table 1 Data format available from Google Finance

Date	Open	High	Low	Close	Volume
15-Dec-08	13.30	13.39	13.23	13.34	144567

In order to obtain a decision making system based on classification techniques a particular company must be selected. This company must satisfy some restrictions:

- It must not be a technological security, in order to avoid using data from year 2000 economic bubble,
- It must not be a financial institution, due to the erratic behaviour of the past years,
- It must have a significant influence within the American stock market,
- It must be a reference company.

Due to these restrictions and after a careful study of the stock market, Alcoa was selected to complete the project. Alcoa is the world's leading producer and manager of primary aluminum, fabricated aluminum and alumina facilities, and is active in all major aspects of the industry. The targeted data includes information from the beginning of 1995 till the end of 2008.

2.1 Financial Indicators

This target data must be processed and transformed into useful data written in terms of financial indicators [3]. The most widely used indicators are based on graphical methods (where price oscillation is studied) and on mathematical methods (where mathematical equations are applied).

The *Simple Moving Average (SMA)* is the unweighted mean of the previous n data points. It is commonly used with time series data to smooth out short-term fluctuations and highlight longer-terms trends and is often used for the analysis of financial data such as stock prices (P), returns or trading volumes. There are various popular values for n, like 10 days (short term), 30 days (intermediate term), 70 days (long term) or 200 days depending on the number of days considered.

$$SMA_{n+1}(P)=\frac{P_n+P_{n-1}+\cdots+P_1}{n}$$

The moving averages are interpreted as support in a rising market or resistance in a falling market. The SMA treats all data points equally and can be disproportionally influenced by old data points. The exponential moving average addresses this point by giving extra weight to more recent data points.

The *Exponential Moving Average (EMA)* applies weighting factors which decreases exponentially, giving much more importance to recent observations while not discarding older observations totally. The degree of weighting decrease is expressed as a constant smoothing factor α, which can be expressed in terms of n.

$$\alpha=\frac{2}{n+1}$$

And the exponential moving average is expressed as:

$$EMA_{n+1}(P)=P_n\alpha+EMA_n(1-\alpha)$$

The moving averages are adequate indicators to determine tendencies. When a short term moving average crosses over a longer term moving average we have a rising tendency. Whereas when a short term moving average crosses under a longer term moving average we have a falling tendency. The moving averages have a drawback since the information provided has a time lag of several days. This can be avoided by using more powerful indicators.

The *Moving Average Convergence Divergence (MACD)* is a combination of two exponential moving averages with different number of data points (days). The *MACD* is calculated by subtracting the y-day *EMA* from the x-day *EMA*. A z-day *EMA* of the *MACD*, called the "signal line", is plotted on top of the *MACD*, functioning as a trigger for buy and sell signals. Variables x, y, and z are the parameters of the *MACD*, usually x = 12, y = 26 and x = 9. This is illustrated in Figure 2, the dark line is the *MACD* whereas the light line is the signal, on the top of the figure the histogram (signal minus *MACD*) is drawn.

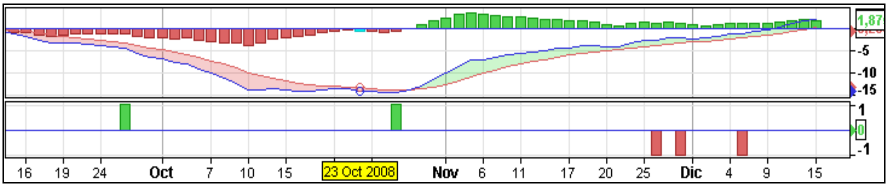


Fig. 2 The Moving Average Convergence Divergence

The *MACD* is a trend-following momentum indicator that shows the relationship between two moving averages of prices. Its two-folded interpretation is as follows. If the *MACD* line and signal line are considered, when the *MACD* falls below the signal line, it may be time to sell. Conversely, when the *MACD* rises above the signal line, the price of the asset is likely to experience upward momentum, it may be time to buy. If the histogram and the price curb are considered, a divergence can be found and buying-selling signals can be obtained.

The *stochastic (K)* is an indicator that finds the range between an asset's high (H) and low price (L) during a given period of time, typically 5 sessions (1 week) or 20 sessions (1 month). The current securities price at closing (C) is then expressed as a percentage of this range with 0% indicating the bottom of the range and 100% indicating the upper limits of the range over the time period covered.

$$K = 100\frac{C - L}{H - L}$$

An interesting interpretation for the stochastic can be obtained if the moving average associated with the stochastic *D(K)* is considered: The crossing points between *K* and *D* trigger buy and sell signals. When *K* crosses over a rising *D* it may be time to buy, conversely when *K* crosses over a falling *D* selling might be convenient. If the stochastic and the price curb are considered, a divergence can be found and buying-selling signals can be obtained.

The representation of the stochastic can be seen in Figure 3.

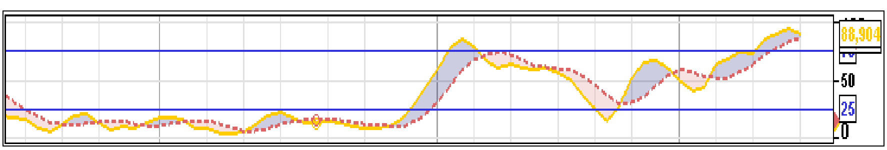


Fig. 3 The Stochastic Indicator (*K*)

The *Bollinger Bands (BB)* consists of a set of three curves drawn in relation to security prices. The middle band is a measure of the intermediate-term trend, usually a moving average (simple or exponential) over n periods. This middle band serves as the base to construct the upper and lower bands. The interval between the upper and lower bands and the middle band is determined by volatility, typically the interval is Q times the standard deviation of the same data that were used to calculate the moving average. The default parameters most commonly used are n = 20 periods and Q = 2 standard deviations.

95% of security prices can be found within the Bollinger Bands, the band represents areas of support and resistance when the market shows no tendencies. When the bands lie close together a period of low volatility in stock price is indicated. When they are far apart a period of high volatility in price is indicated.

Generally Bollinger Bands are used together with other indicators to reinforce its validity. However, on its own they can trigger buy and sell signals. When prices touch the lower band it might be convenient to buy, conversely when prices touch the upper band selling might be convenient.

2.2 Processing of the Financial Indicators

The financial indicators introduced above will aid us in the construction of our Object Attribute Table from the raw data base. In order to calculate these indicators, R a language for statistical computing and graphics, is used [4]. R is a popular programming language used by a growing number of data analysts inside corporations and academia. R is applied mainly as a statistical package but it can be used in other areas of science such as numerical analysis, signal processing, computer graphics and so on because of the number of growing implemented libraries in R to these scientific areas. See [5] for details on the use of R in academia and other institutions. R is available as Free Software under the terms of the Free Software Foundation's GNU General Public in source code form. It compiles and runs on a wide variety of UNIX platforms and similar systems (including FreeBSDband Linux, Windows and MacOS).

The financial indicators used in this paper are calculated by the application of R's TTR package to the raw data retrieved from Google finance. These indicators must now be transformed into useful data as follows.

2.3 Transformed Data: The OAT

The next step in the data processing phase is the generation of the Object Attribute Table [6, 7]. The indicators considered are the following: *MACD* (Moving Average Convergence Divergence), *EMA(C)* (Exponential Moving Average of the Closing Price), *EMA(V)* (Exponential Moving Average of the Volume), *K* (Stochastic) and *BB* (Bollinger Bands) and they generate eight binary attributes A1, A2, A3, A4, A5, A6, A7 and A8 in the following manner.

- The *MACD* indicator identifies buying (B) or selling (S) orders depending on the crossing of intermediate term moving averages. Two binary attributes A1 and A2 are used as follows: if an S signal is generated then A1= 1; if a B signal is generated then A2= 1.
- The *EMA(C)* indicator when compared to the stock's actual price generates attribute A3, if the asset's ranking is strong then A3=1.

- The *EMA(V)* indicator when compared to the actual volume generates attribute A4, if the asset's reliability increases then A4= 1.
- The *K* indicator also identifies buying (B) or selling (S) orders depending on the crossing of moving averages, in particular *K* and *D*. Two binary attributes A5 and A6 are used as follows: if an S signal is generated then A5= 1; if a B signal is generated then A6= 1.
- The *BB* indicator generates 4 zones. Two binary attributes A7 and A8 are used as follows: if the asset's ranking is strong then A7= 1 and A8=1; if the asset's ranking is weak then A7= 0 and A8=0; if a B hint is found then A7= 0 and A8=1; if an S hint is obtained then A7= 1 and A8=0.

Thus, the OAT of transformed data has 8 columns corresponding to the above mentioned binary attributes and a total of 3515 rows corresponding to available transformed financial data, as illustrated in Table 2. The data contained in the table must be filtered. First of all, repeated rows are eliminated. Secondly, incomplete rows are also removed. The remaining table has 61 rows. The next step is to assign the class to each of the remaining rows taking into account the values of the attributes as indicated by the financial expert. Three different classes are used depending on whether buying (B), selling (S) or no action (N) must be taken.

Table 2 Object Attribute Table

A1	A2	A3	A4	A5	A6	A7	A8	class
0	0	0	0	1	0	1	0	S
0	1	0	0	0	0	0	1	B
0	0	1	0	0	1	0	1	N
0	0	0	1	0	1	1	0	N
1	0	1	0	0	0	0	1	S
1	0	1	1	0	0	1	0	S
…	…	…	…	…	…	…	…	…

3 The Data Mining Phase

In the data mining phase, the data contained in the OAT is converted into useful patterns. In particular, a decision tree will be obtained. The decision tree is one of the most popular classification algorithms. A decision tree can be viewed as a graphical representation of a reasoning process and represents a sequence of decisions, interrelationships between alternatives and possible outcomes. There are different trees available depending on the evaluation functions considered. In this project, the ID3 algorithm is used due to its simplicity and speed [8]. ID3 is based on information theory to determine the most informative attribute at each step of the process. In this sense, it uses information gain to measure how well a given attribute separates the training examples into the classes.

Figure 4 shows the resulting decision tree. The decision tree has a total of 37 nodes, 18 corresponding to the 8 attributes used and 19 leaf nodes or branches. The average branch length is 6.

Attributes A1, A2 and A5 appear at the root of the tree suggesting the importance of indicators *MACD* and *K* in the classifying process. This is in accordance with the experts that consider *MACD* and *K* as key indicators in the buying-selling decisions, whereas *EMA(C)*, *EMA(V)* and *BB* are used as reinforcement indicators.

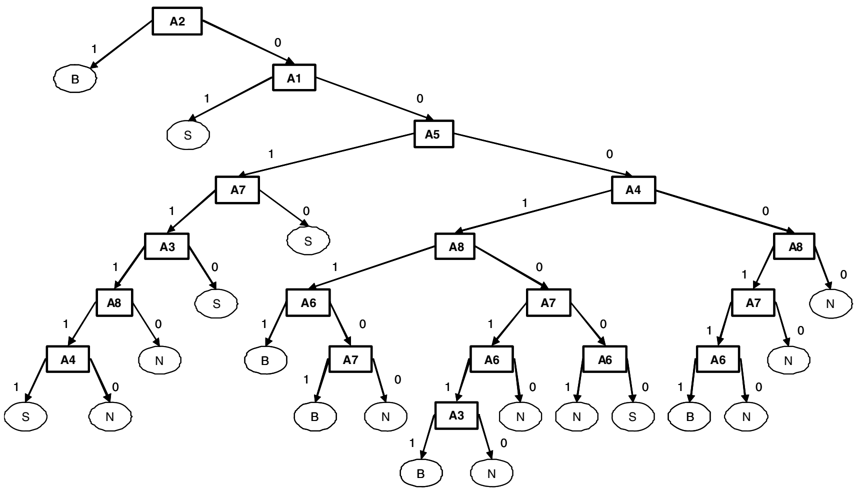


Fig. 4 The ID3 Decision Tree

4 The Evaluation Phase

The evaluation phase proves the consistency of the ID3 decision tree by means of a testing set. The testing set is formed by the stock market prices of a different asset, of similar characteristics to the one used to train the tree. This raw data are transformed into a testing OAT using the previous indicators and attributes. Now the decision tree is applied to the testing OAT and all instances are classified. These classes obtained with the tree are compared with the classes corresponding to the instances of the testing OAT as suggested by the expert. The evaluation of the method provides the results shown on Table 3.

Table 3 Evaluation of the method

Parameters	3-class tree
Correct decision	113
Incorrect decision	176
Correct decision (%)	39.10
Incorrect decision (%)	60.90

These results might seem poor. However, stock market analysis is an unpredictable field and even though accuracy is desired, profit associated to capital gain is crucial. Therefore, classification accuracy is not the only parameter that must be evaluated. When profit is considered, the method provides the following results: if an instance is correctly classified by the tree, the percentage gain is 579.71%, if an instance is not correctly classified by the tree, the percentage loss is -461.56%. This produces a total yield of 118.15% which is an acceptable result.

5 Conclusions and Future Work

The stock market is an unpredictable, erratic and changeable domain. Nevertheless, tools that predict the stock market's behaviour exist. This paper considers a method based on financial indicators to generate a decision tree that classifies buying-selling orders. A decision tree in Artificial Intelligence is equivalent to an automatic investment system in the stock market analysis. The procedure developed here is long and costly, but once the tree is generated, it's automatic and applicable to any financial asset.

The initial raw data are daily stock market values of Alcoa, these data are transformed into an OAT and used as a training set to construct a decision tree by means of the ID3 algorithm. This decision tree has been tested and evaluated. The results show a 40% accuracy percentage and a 118% profit gain that are considered satisfactory.

Currently, a second OAT with only two classes, buying (B) and selling (S) is being studied. In this case the no action class (N) has not been considered. This 2-class decision tree will allow for a more complete evaluation of the method.

This paper offers a limited vision of one of the many solutions available. The following aspects can be considered as future work: other different financial indicators may offer better results and should be studied, also other decision trees, such as C4.5 or J48, can be obtained using different learning algorithms and should be evaluated. The stock market is a dynamic world, and the decision tree could present time-related disturbances. Therefore, an autonomous system capable of dealing with these changes should be considered.

Acknowledgements. The authors want to thank Damià Vaquer, Gerard Draper and Jaume Jaume for their helpful comments and suggestions.

References

1. Fayyed, U., Piatetsky-Shapiro, G., Smyth, P.: From Data Mining to Knowledge Discovery in Databases. American Association for Artificial Intelligence, AI Magazine 96, 37–54 (Fall 1996)
2. Fiol-Roig, G.: UIB-IK: A Computer System for Decision Trees Induction. In: Raś, Z.W., Skowron, A. (eds.) ISMIS 1999. LNCS, vol. 1609, pp. 601–611. Springer, Heidelberg (1999)

3. Weinstein, S.: Stan's Weinstein's Secrets For Profiting in Bull and Bear Markets. McGraw-Hill, New York (1988)
4. The R project for Statistical Computing, http://www.r-project.org/
5. http://www.nytimes.com/2009/01/07/technology/business-computing/07program.html
6. Miró-Julià, M.: Knowledge Discovery in Databases Using Multivalued Array Algebra. In: Moreno-Díaz, R., Pichler, F., Quesada-Arencibia, A. (eds.) Computer Aided Systems Theory - EUROCAST 2009. LNCS, vol. 5717, pp. 17–24. Springer, Heidelberg (2009)
7. Fiol-Roig, G.: Learning from Incompletely Specified Object Attribute Tables with Continuous Attributes. Frontiers in Artificial Intelligence and Applications, vol. 113, pp. 145–152 (2004)
8. Quinlan, J.R.: Induction of decision trees. Machine Learning 1, 81–106 (1986)

Synthesis and Analysis of Classifiers Based on Generalized Model of Identification

M. Tatur, D. Adzinets, M. Lukashevich, and S. Bairak

Abstract. In this paper we propose a generalized model of identification which displays flexible transformation within the framework of generally known paradigms by changing tunings. The application of this model enables to synthesize various classifiers using a priori information about definite applied tasks of identification. So, we describe the approach to the solution of the problem of generation of representative training sequences and correct comparative evaluation of classifiers.

Keywords: Identification, Classifier, Neuron Model, Fuzzy Logic.

1 Introduction

The efficiency of a recognition decision-making system directly depends on the classifier and the learning algorithm. The learning algorithm is an intellectual core of a system taken as a whole. There exist a number of classification methods in modern recognition theory (minimum distance classifier, neural networks, support vector machine, etc.) with relevant learning algorithms and set of samples of their application [1-4]. Developing a classifier for any applied task a researcher has to perform the following operations:

1. choose a classification method (the structure of a classifier).
2. generate representative training samples.
3. find and apply an effective learning method.
4. estimate the efficiency of the technical implementation.

A classification problem is usually considered as the division of spots in the hyperspace of informative features, and a learning algorithm – as a construction of

M. Tatur · D. Adzinets
Intellectual Processor Ltd,
22/2-310, Colasa str., Minsk, Belarus

M. Lukashevich · S. Bairak
Belarusian State University of Informatics and Radioelectronics,
6, Brovki str., Minsk, Belarus
e-mail: {tatur,lukashevich}@bsuir.by

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 529–536.
springerlink.com

dividing hypersurface. The most "simple" problems are linearly separable, i.e. for their solution it is enough to have a surface described by a linear polynom. Difficult "problems" require dividing surfaces with polynoms of higher order for their solution. In this case multilayer neural networks with developed mathematical apparatus are practically the only approach to the problem solution. The neural network approach is very popular. But it has a number of restriction such as the problem of training convergence, conformity of neural network structure to a solved problem, training and classification interpretation. Logical and fuzzy conclusion methods, decision trees that are also applied to classification and decision-making problems and operate with extracted knowledge and therefore are easily interpreted. However in the recognition theory these methods are considered separately and alternatively. There is a terminological discrepancy. For example, the concepts of a neural layer in classical and fuzzy neural networks are different, because neuron models are non-comparable. The efforts to develop a neuron hybrid model have been reported. However the given technical results do not show a flexible conversion from one method to another [5-7].

The classifier is an abstract automatic machine (mathematical model). It makes a decision to what class the vector of informative features belongs. It is supposed that informative features are taken into account and combined according to some rule in to an integrated parameter. This parameter allows to making a decision. The most trivial (elementary) decision is the dual decision – YES/NO. Thus, the ordinary classifier is the classifier, which makes the simplest decision about belonging to the given class. In the literature, this kind of classification is called identification. In practice a multiclass problem is often a set of identification problems. It is very convenient for its performance evaluation, criteria definition, interpretation of classification results. Therefore, in some studies theorems, statements etc. are proved and illustrated for identification problems, and then generalized for multiclass problems. The diagram for multiclass problem as a set of independent identification problems is presented in Fig. 1. Further in this article mathematical models of classifiers for identification problems will be considered.

Training of classifier, in a broad sense, embraces practically all classification methods and includes various aspects of this issue, such as learning algorithms, a

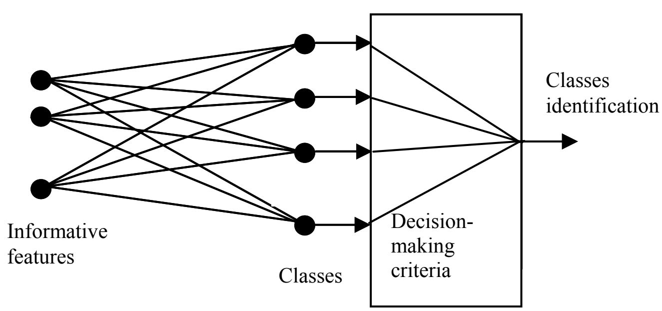


Fig. 1 General scheme of classification

selection and tuning of decision-making criteria, formation of representative training data sets, and evaluation of training efficiency.

Uniform databases are used for correct comparative estimations of complete recognition systems (FERET database, texture databases, medical diagnoses databases, etc.) [8-10]. The abstract data are necessary for testing and evaluation of abstract classifiers. Further, *a learning set* is a base of numerical vectors with familiar responses. There are no conventional numerical databases with conventional experiments for testing of classifier models and learning algorithms.

The original identifier model is presented in this article. It reminds a formal neural model and combines the basic properties of classical and fuzzy prototypes. This model allows to synthesize classifiers adequately for applications, to solve generation problems of representative learning sequences and correct comparative evaluation of final technical realizations.

2 Concept and Justification for Mathematical Model

We introduce a concept of the *ideal classifier* for the explanation of the suggested approach. The ideal classifier is a decision function in the identification task

$$Z = F(X,Y), \tag{1}$$

where **X** - vector of input informative features, **Y** – vector of constants.

Constants are installation-specific settings for specific solution. Functional **F** and settings **Y** are a priori known.

In the development of a classical classifier it is necessary to select a functional **Ψ** and settings **U** with minimum mean square error for learning and testing succession.

$$Z^* = \psi(X,U) \tag{2}$$

$$d = \sqrt{\frac{1}{(m-1)}\left(\frac{\sum (z_i - z_i^*)^2}{\sum z_i^2}\right)} \tag{3}$$

where **m** – is a number of test vectors.

In a considerable number of applied tasks classification function can be written with a known functional **F** and inaccurately given settings of **Y`**.

$$Z' = F(X,Y') \tag{4}$$

It is unnecessary to choose a mathematical method or structure of a classifier (for example, neural network structures). Training of a classifier is a customization or specification of parameters **Y`**.

We consider construction stages and model substantiations using series of simple examples with four standard identification tasks, ranking by the increase of «intellectuality degrees».

1. The simplest variant of identification task is the task of luggage sorting on total value of dimensions: length, width and height. All three measured parameters are equivalent, i.e. equally influence decision-making. The usual threshold function is used as decision making function. Designate "preparation" function for decision-making as **S (X)**, and the set of informative features as **X (1 ÷ n)**. Then:

$$Z = \begin{cases} 1, S(X) > p \\ 0, S(X) < p \end{cases} \tag{5}$$

where $S(X) = \sum_{i=1}^{n} x_i$, **p** – defined threshold.

It is obvious that in cases when the sum of measurements aspires to a threshold, **S (X) → p** the probability of wrong decision-making increases because there are errors of the first and the second kind. Traditional solutions of the given problem are: the introduction of an ejection subclass in a threshold vicinity **p** ± Δ (when the solution is not accepted) and the usage of fuzzy solution. In case of a fuzzy decision-making (linear-approximated function of a fuzzy threshold), function **Z** looks like in Fig. 2:

$$Z = \begin{cases} 1, S(X) > p_2 \\ \dfrac{S(X) - p_1}{p_2 - p_1}, p_1 \le S(X) \le p_2 \\ 0, S(X) < p_1 \end{cases} \tag{6}$$

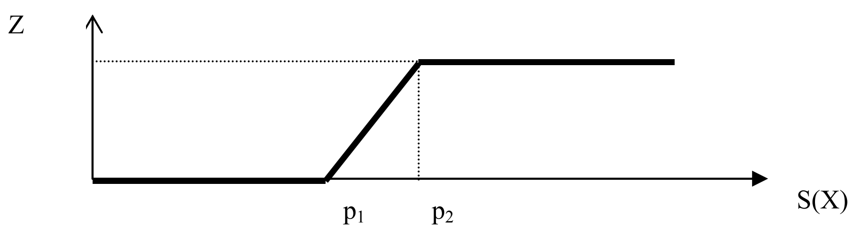


Fig. 2 The linear-approximated decision function

2. Complicate the initial task of object sorting. Assume that informative features **X** have various physical sense and a different degree of influence on decision-making process. Than various object characteristics (for example, weight, size, humidity etc.), as a result of expert estimations or training process, can be reduced to an integrated measure using linear weight coefficients **W (1 ÷ n)**. Informative features can be both promoting and interfering for decision-making. If it is true for developed recognition system then it is considered by **Sign(w_i)**.

$$S(X)=\sum_{i=1}^{n} w_i x_i \tag{7}$$

3. Continue to complicate the task. Let informative features $\boldsymbol{x_i}$ for object be parameterized. They should be in breaking bounds. These bounds can be accurate fixed (in that specific case) and fuzzy (in a general case) using functions **φ ($\mathbf{x_i,a_i,b_i,c_i,d_i}$)**. For example, at person of identification by photograph the informative feature «distance between mouth corners» has a rather wide range, and «distance between pupils» - practically does not vary.

Select linear-approximated function for fuzzy definition of parameters and receive (Fig. 3):

$$S(X)=\sum_{i=1}^{n} w_i \varphi(x_i,a_i,b_i,c_i,d_i) \tag{8}$$

where $\varphi(x_i,a_i,b_i,c_i,d_i)=\begin{cases} 0, a_i < x_i, x_i > d_i \\ \dfrac{d_i - x_i}{d_i - c_i}, c_i \le x_i \le d_i \\ 1, b_i < x_i < c_i \\ \dfrac{x_i - a_i}{b_i - a_i}, a_i \le x_i \le b_i \end{cases}$

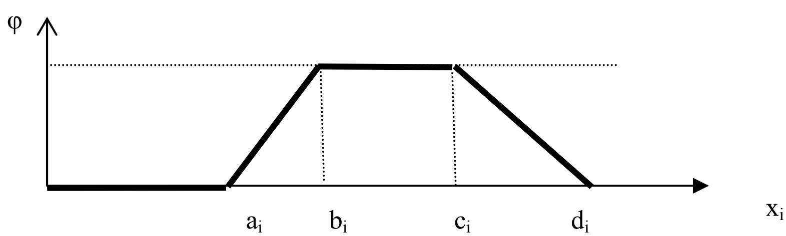


Fig. 3 Piecewise-linear approximation of parameterization function of an informative features

4. And once again we complicate the task. In some cases decision-making takes into account additional logical conditions by which informative features are connected. For example, the condition that all or key informative features (set by subset **N** ') were in certain tolerance of parameters is often applied. For example, at person of identification by photograph, the informative feature «distance between pupils» is one of key parameters. It means that if it appears not in tolerance, then $\mathbf{L} \rightarrow \mathbf{0}$, and value of total decision function $\mathbf{Z^L} \rightarrow \mathbf{0}$, irrespective of the hit rate of other informative features.

$$L = \min_{i=1}^{n} \varphi'(x_i, a_i, b_i, c_i, d_i) \quad (9)$$

where $\varphi'(x_i, a_i, b_i, c_i, d_i) = \begin{cases} \varphi(x_i, a_i, b_i, c_i, d_i), i \in N' \\ 1, i \notin N' \end{cases}$

$$Z^L = LZ \quad (10)$$

Thus, in expressions (1, 5-10) the generalized mathematical identification model is described. Mathematical notation can be presented graphically using the form of a structural model (Fig. 4). All model settings **W, A, B, C, D , N', $\mathbf{p_1}$, $\mathbf{p_2}$** are a subset of vector Y according to accepted denotations.

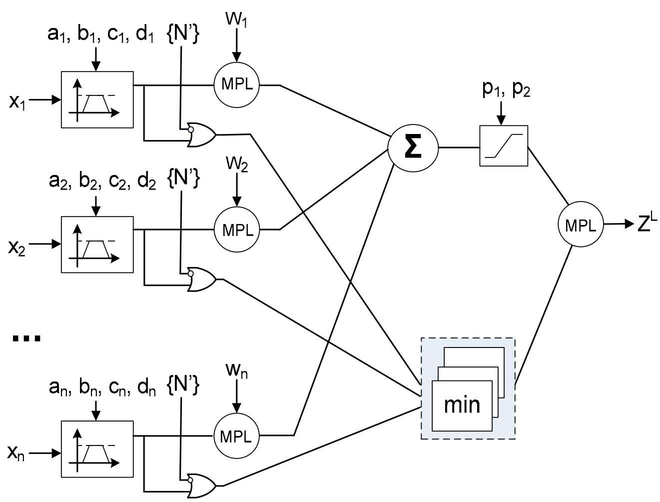


Fig. 4 Graphics representation of the generalized model of the elementary classifier

3 Properties of Model

Methodological model significance is a combination of accurate and fuzzy, arithmetic and logic components, which can be perform both independently or jointly. On the hole, the model connects two directions known as data processing and knowledge processing. The model shows smooth **transformation** of qualitative distinctions and allows to form typical adequate functions for applied recognition problems by a combination of options of parameters **W, A, B, C, D**, **N'**, $\mathbf{p_1}$, $\mathbf{p_2}$. For example:

$\mathbf{p_1 = p_2}$ - realization of binary decision-making;
$\mathbf{p_1 \neq p_2}$ - realization of fuzzy decision-making;
$\mathbf{w_i = 1}$ for all **i** – realization of absence of weighing of features;

$wi \neq 1$ - realization of the weighed summation of features;

$\mathbf{a_i = b_i = 0, c_i = d_i = 1}$ – realization of absence of parametrization of informative features;

$\mathbf{a_i = b_i, c_i = d = 1}$ - realization of a clear boundary of parameters for each informative feature;

$\mathbf{a_i \neq b_i, c_i \neq d_i}$ - realization of indistinct borders of parameters for each informative feature;

$\mathbf{a_i = bi = c_i = d_i}$ – realization of arithmetic or logic coincidence of an informative feature with parameter-standard;

if $\mathbf{N'} \in \varnothing$ – logic conditions are not used;

if $\mathbf{N'} \notin \varnothing$ - realization of logic conditions on key features, depending on parameterization to each informative feature;

$\mathbf{p_1 = p_2 = 0}$, **N'** - nonempty set - realization of logic conditions only.

The further model development can be carried out by the way of complicating a logic function. (In the present work it is not considered).

The suggested approach allows generate numerical databases **X→Z** for typical classification problems. These databases can be applied for testing the efficiency of classifier and training algorithms, irrespective of a mathematical method of realization. The term «ideal base» is used because exact values of function **Z=F (X, Y)** are calculated, if we set parameters of model adjustment - **Y** and generate consequence of the input data - **X**. Traditionally applied, as a rule, bases belong to subject domains (for example, images). They receive on the basis of expert estimations. Creating a database, it is possible to set purposefully: the number of informative features -**n**; the number of test vectors -**m**; a format and accuracy of presentation of input and output information; brackets for data normalization; a kind and level of noise.

Test bases can be used in two different applications.

In the first application: for verification and performance evaluation of *the define* classifier and training algorithm, developed for the define recognition system; for testing the classifier and training algorithm to "expert noises" tolerance. In this application it is necessary to generate a test database to the define recognition model. While conducting of experiments the *particular methods* of comparative analysis can be used.

In the second application: for creation of the united system of objective `estimation and realization of competitive selection of classifiers and training algorithms; for creation the library of classifiers and training algorithms, which reflect the current level of knowledge in this area.

4 Conclusion

In the study the generalized identification model is presented. Model functionality includes both the operation of weighted summation characteristic of classical formal neuron and elements of fuzzy output characteristic of fuzzy neural networks. In the presented model it impossible to consider a functional of the applied task using model settings. On the basis of model and the presented examples of four standard tasks of identification it is possible to synthesize the classifiers which can be adequate to applied tasks (under a condition of a priori a known functional). If the functional is a priori unknown, the construction task of the classifier on complexity is comparable to the choice task of a multilayered neural network structure. The structure of the offered model in comparison with a multilayered neural network is adequately interpreted by researchers and developers. It can be used for its purposeful development and training.

If argument values are set as **X, W, A, B, C, D**, **N'**, $\mathbf{p_1}$, $\mathbf{p_2}$, it is possible to calculate ideal function values **Z=F(X,Y)**. Thus, the model allows to create (generate) test numerical databases for typical classification problems. It is possible to carry out the correct comparative analysis of classifiers and learning algorithms, choose necessary classification methods and develop and verify algorithms of training using such "ideal" data sets.

The generalized character of the model and possibility of a flexible functional adaptation by options allow to consider it as a mathematical basis for realization of mass operations by working out of perspective neuro-computer architecture.

References

1. Yu, J., Ou, Y., Zhang, C., Zhang, S.: Identifying Interesting Visitors through Web Log Classification. Intelligent Systems 20(3), 55–59 (2005)
2. Krisnapuram, B., Hartemink, J., Carin, L., Figneiredo, M.A.: A Bayesian Approach to Joint Feature Selection and Classifier Design. Pattern Analysis and Machine Intelligence 26(9), 1105–1111 (2004)
3. Melnik, O., Vardi, Y., Zhang, C.-H.: Mixed Group Ranks: Preference and Confidence in Classifier Combination. Pattern Analysis and Machine Intelligence 26(8), 973–981 (2004)
4. Sun, Z., Wang, Y., Tan, T., Cui, J.: Improving Iris Recognition Accuracy via Cascaded Classifiers. In: Zhang, D., Jain, A.K. (eds.) ICBA 2004. LNCS, vol. 3072, pp. 418–425. Springer, Heidelberg (2004)
5. Theodoridis, S., Koutroumbas, K.: Pattern Recognition, 2nd edn. Academic Press an imprint of Elsevier, USA (2003)
6. Yager, R.: Participatory Learning: A Paradigm for More Human Like Learning. In: Proceedings of IEEE International Conference on Fuzzy Systems, Budapest, IN 0009-1044 (2004)
7. Chou, C.-H., Lin, C.-C., Liu, Y.-H., Chang, F.: A prototype classification method and its use in a hybrid solution for multiclass pattern recognition. Pattern Recognition 39(4), 624–634 (2006)
8. Phillips, P.J., Wechsler, H., Huang, J., Rauss, P.: The FERET Database and Evaluation Procedure for Face Recognition Algorithms. Image and Vision Computing Journal 16(5), 295–306 (1998)
9. Qiu, X., Sun, Z., Tan, T.: Global Texture Analysis of Iris Images for Ethnic Classification. In: Zhang, D., Jain, A.K. (eds.) ICB 2005. LNCS, vol. 3832, pp. 411–418. Springer, Heidelberg (2005)
10. Tung, W., Quec, C.: GenSo-FDSS: a neural-fuzzy decision support system for pediatric ALL cancer subtype identification using gene expression data. Artificial Intelligence in Medicine 33, 61–88 (2005)

Multiagent Based Spectrum Sharing Using Petri Nets

Usama Mir, Leila Merghem-Boulahia, and Dominique Gaïti

Abstract. This paper presents a cooperative framework for dynamic spectrum access in cognitive radio networks, where the cognitive radio devices may use the currently unutilized spectrum dynamically and opportunistically. The proposed solution works on the principle of multiagent system that allows the cooperation amongst several participating devices. The key aspect of our design is the deployment of agents on each of the primary and cognitive radio device that cooperates in order to have a better and dynamic use of the spectrum. For cooperation, contract net protocol is considered allowing spectrum to be dynamically allocated by having a series of exchange of messages. Due to the concurrent, distributed and autonomous nature of our approach, Petri nets are adopted to model the behavior of cooperative framework. Based on Petri nets, we study the internal message passing and spectrum allocation mechanism between the participating primary and cognitive radio users and identify the information needed by the agents to make cooperative decisions. Then, using empirical results, we show that our approach achieves up to 80% of the whole utility within the span of few messages, and provides an effective mechanism for distributed spectrum allocation.

Keywords: Multiagent Systems, Cognitive Radio, Dynamic Spectrum Sharing/allocation, Cooperation, Petri Nets.

1 Introduction

The current increase of traffic in wireless systems has created a paradigm shift of spectrum shortage [12]. This shortage problem comes from the fact that current spectrum assignment policies are static, in which the spectrum is allocated to a user or a service provider for a specific time period on long-term basis. While according to a study by the Federal Communications Commission (FCC) [2], in all the areas (either rural or urban), utilization of assigned spectrum can be as low as 5-10%, thus the latest wireless technologies are forced to utilize unlicensed spectrum bands. This phenomenon results in inefficient spectrum assignments with the higher possibility of device level interference and spectrum wastage.

Usama Mir · Leila Merghem-Boulahia · Dominique Gaïti
ICD/ERA, FRE CNRS 2848, Université de Technologie de Troyes, 12 rue Marie Curie, 10010 Troyes Cedex, France
e-mail: {usama.mir,leila.merghem_boulahia,dominique.gaiti}@utt.fr

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 537–546.
springerlink.com

The necessary and appropriate solution to the above problem is dynamic spectrum allocation. In this new concept, devices using cognitive radio[1] [5] will be able to detect and use the empty spectrum portions opportunistically, without interrupting the working of legacy primary (or licensed) users. However, one of the key issues in cognitive radio (CR) networks is how to achieve efficient (and dynamic) spectrum sharing among the nodes, while avoiding inter-device collisions. A non-cooperative node can cause harmful interference to its neighbors [13] and hence can reduce spectrum utilization.

Similar to the distributed nature of CR networks, a multiagent system (MAS) [11, 13] is composed of autonomous agents that communicate and perform their functions without a central control. Each agent can be regarded as an autonomous entity that governs the operation of the system on which the agent is embarked. Like CR nodes, agents work dynamically to fulfill their user needs and no single agent has a global view of the network. Moreover, in MAS, cooperation is mutually beneficial for the participating agents, even if they are selfish. Among several models of agents' interactions, one can find contract net protocol (CNP) [4] as one of the effective way of agents' cooperation and resource allocation. Its decentralized structure based on the announcement of tasks, receiving of bids, and assigning of contracts makes it highly suitable for spectrum allocations in CR networks.

Therefore, in this paper, an MAS based solution is proposed to mitigate the effects of static spectrum assignments. Specifically, in our approach, the agents are embarked over each of the primary and secondary user devices. In order to cooperate [10], a secondary user (SU)[2] agent should send messages to the appropriate neighboring primary user (PU)[3] [12] agents whenever needed and, subsequently, the related PU agents should reply to these agents in order to make a spectrum sharing agreement. We propose that the SU agents should take their decisions based on the amount of spectrum, time and price proposed by PU agents [10] and should start spectrum sharing whenever they find an appropriate offer. Then, after completely utilizing the desired spectrum, SU agents should pay the agreed price to the respected PU agents. To facilitate the feasibility analysis and to show the internal behavior of our cooperative framework, Petri net (PN) [1, 4] models are proposed to capture the frequent interactions between primary and secondary user agents. Its graphical representations via firing of several tokens can help us visualizing the allocation of a spectrum resource from one agent to the other. Based on PN model, we study the messages exchanged by agents to make cooperative decisions, while passing through several cooperative stages. In this context, our prior work [7] proposed a design model for an SU agent with interlined working of various modules. It also details the working of the whole system using various algorithms. However, it lacks the complete analysis in order to show the correct inter-node cooperative behavior. Thus, in this work we present a dynamic and cooperative spectrum sharing framework with its feasibility analysis using Petri nets.

---

[1] Is an intelligent wireless communication system that senses the available empty spectrum portions (bands) and provides the means for sharing these available bands with the neighboring cognitive radio users [8].

[2] Secondary (unlicensed or cognitive radio) users are those which do not have license/rights to completely utilize the spectrum.

[3] Referred to as the legacy or licensed users having full rights for spectrum access.

The rest of this paper is organized as follows. The related work is presented in Section 2. Section 3 presents our cooperative approach. Analysis of the cooperative framework is performed in Section 4. Section 5 presents some numerical results Section 6 summarizes the paper with a conclusion and future perspectives.

2 Related Work

The spectrum sharing has recently attracted the research attention especially using game-theoretical approaches [3, 9], but there is not much work done for the utilization of cooperative MAS in CR netowrks. A related approach is proposed in [5], in which an MAS is used for information sharing and spectrum assignments. All the participating agents deployed over access points (APs), form an interacting MAS, which is responsible for managing radio resources across collocated WLANs. Furthermore, each agent is responsible for making the main decisions of AP, including the number of requests to accept from mobile stations, controlling the device's transmission power along with the best channel selection. The authors have not provided any of the algorithms and results for their proposed approach. In [6], another MAS based approach is presented, where a distributed and dynamic billing, pricing and resource allocation mechanism is proposed. The agents work as the *auctioneers* and the *bidders* to share the spectrum dynamically. The complete statistical analysis of the approach are provided with the calculation of CR user's utility, input strategy, cost, but still the whole system is really complex, including the complexity in time required to calculate the optimal spectrum allocation.

3 Cooperative Spectrum Sharing

3.1 Agent

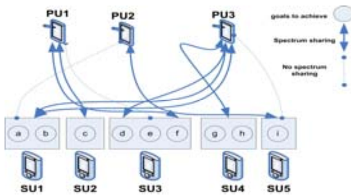
An agent is a dynamic and loosely coupled unit, having the capabilities of performing a task autonomously, based on the knowledge received from its environment and/or through other agents' interactions [11]. Generally, an agent is appropriate for an SU node in a sense that it allows the introduction of various artificial intelligence (AI) techniques to CR networks and helps an SU node to cooperate with its neighboring devices.

3.2 Description of Cooperative Framework

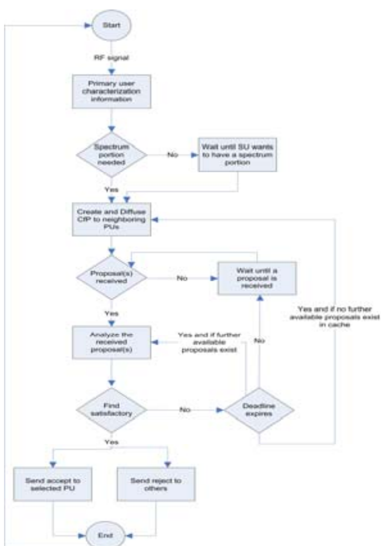
We consider the spectrum allocation challenges in a private area or a well identified administrated perimeter such as a university campus, a building or a conference hall. There are subsets of ad-hoc WLANs with the agents deployed at each of the primary and secondary user devices for the purposes of cooperation. Whenever an SU needs spectrum, its agent starts communicating with the relative PU agent (having that empty spectrum part), until a spectrum sharing agreement is made.

To simplify and to better explain our approach, an ad-hoc environment with three PUs and five SUs is considered. Each of the PUs has several empty portions of the spectrum for which there is some associated price. To allocate the spectrum portions dynamically and to maintain a cooperative mechanism between primary and secondary user devices, message passing of contract net protocol (CNP) is used, where the SU agents are considered as the managers and PU agents work as the potential contractors. Related to each SU, there are some goals to achieve (or tasks to perform) such as to have video conferencing on mobile devices, watch high multimedia videos, transfer high resolution advertisements, etc. To fulfill these goals, the SU agents need the spectrum portions from the related PU agents.

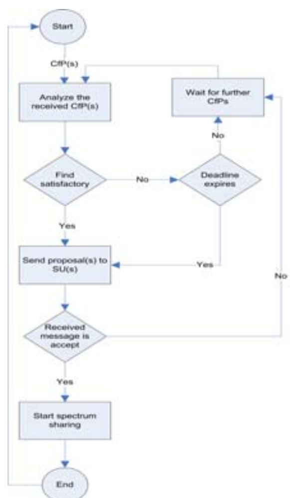
Figure 1(a) shows the cooperation between primary and secondary users with the respected spectrum allocations. In this figure, the numbered circles denote the desired goals (e.g. running multimedia applications, watching videos, etc) for



(a)



(b)



(b)

Fig. 1 (a). Spectrum allocations/assignments **(b).** Internal behavior of an SU **(c).** Internal behavior of a PU which an SU wants to have spectrum usage. Each SU agent can send call for proposals (CfPs) to multiple primary users and at the same several primary users can respond via proposals. Moreover, the SU agents diffuse the message to those neighbors which are known via spectrum sensing module of CR [7] and inter-agents' knowledge information [7]. The corresponding figure also represents the accepted and rejected spectrum sharing agreements. SU_1 has made an agreement with PU_3 for goals *a* and *b* respectively, while it is not able to establish the agreement with PU_2 due to its high requirements. SU_3 has established agreements with PU_2, and PU_3 for the goals *d* and *f*, but at the same time, there is no agreement with PU_1 (as shown by a dotted line), for the goal *e*. Last part of figure 1(a) shows the contract between SU_5 and PU_2 for goal *i* and a disagreement with PU_3, due to the lack of satisfaction (shown by SU_5). Slightly less values of parameters such as price, amount of spectrum needed and spectrum holding time are the reasons of these disagreements.

Figures 1(b) and 1(c) show the behaviors of a primary and a secondary user. The behavior follows the same *analyzing*, *characterizing* and *interacting* phases as mentioned in [7]. For an SU, the spectrum sharing process starts by getting the characterization results and its requirements. This process continues until the sending of CfPs, receiving of proposals and ends either by accepting or rejecting a proposal and going back to the start. The PU follows the same pattern by starting its process with the CfP reception and ending it either by an accept or a reject message from an SU.

4 Representation of Cooperative Network Using Petri Nets

To model the cooperative spectrum sharing between primary and secondary users and to capture their behaviors, Petri net (PN) model is proposed. The reason for using PNs is their discrete and distributed nature to capture the dynamics of decentralized networks. PN is a graphical tool for the formal description of the flow of activities in complex systems. Generally, PNs are used to represent the logical interactions among nodes, devices and parts of a system. It has two types of nodes, namely a set of places $P = \{p_1, p_2,\dots. p_n\}$ and a set of transitions $T = \{t_1, t_2, \dots. t_m\}$. A place is represented by a circle and a transition by a bar (or by a box). Further, PNs consist of set of inputs *I*, outputs *O* and the markings *M* (assignment of tokens to the places). The marking of a PN is a vector, the components of which are positive integer values. The dimension of this vector is equal to the number of places. A *token* (represented by a small filled circle) is moved from one place to another when a transition is fired, for example: firing a transition will allocate a resource from one node to another and this firing can be represented by removal of a *token* from input place and its addition into the output place.

Cooperation Model

The proposed Petri net *N* is a five tuple $N = \{P, T, I, O, M\}$, where *P* is the finite set of places, *T* is a finite set of transitions, *I* and *O*, are the input and output functions which specify the input and output places of transitions respectively. $M = \{M_0, M_1, M_2,\dots,M_f\}$ is the set of markings such as M_o and M_f denote the sets of initial and final markings, after firing of all the transitions. Here, we also denote by:

- **t* is the input place *p* of transition *t*, such that *(p,t)* ∈ *I*.
- *t** is the output place *p* of transition *t*, such that *(t,p)* ∈ *O*.
- **p* is the input transition *t* of place *p*, such that *(t,p)* ∈ *I*.
- *p** is the output transition *t* of place *p*, such that *(p,t)* ∈ *O*.

The places represent various states of primary and secondary users and a transition is said to be enabled when an event is about to occur (such as sending message, receiving responses). If a transition $t \in T$ is enabled, then it can be fired. Firing a transition will remove a token(s) from each **t* and will add token(s) to *t**. Formally, firing transitions consists of transforming the M_o of N into M_f as follows:

$$M_f(p) = \begin{cases} M_o(p) - \mathit{firedtoken}(s) & \text{if } p \in {}^*t \\ M_o(p) + \mathit{firedtoken}(s) & \text{if } p \in t^* \\ M_o(p) & \text{otherwise} \end{cases} \tag{1}$$

Figure 2 describes the situation where the SU_5 (of figure 1a) has simultaneously sent its CfPs to PU_2 and PU_3 and it has been able to make a spectrum sharing deal with PU_2. This PN based model is been proposed to show the behavior of a secondary user when it has to deal with more than one proposal. Similarly, the primary user will behave on the same manner when it has to handle more than one CfP. Table 1 shows various states, while table 2 depicts the initial and final markings of tokens after firing of all the transitions. It is clear from table 2 that the final marking M_f becomes '3', only when a resource is been shared or the price is been paid. Before going into details, we can define a few important terms related to our proposed model.

Definition 1. For a secondary user $SU_i \in SU$, the Petri net $N_i = \{P_i, T_i, I_i, O_i, M_i\}$ for an empty part of the spectrum s_i for time t_i, is a working model if and only if the movement from initial marking M_{osi} to final marking M_{fsi}, after firing all transitions, results in the maximization of its utility function U_{sui} such that $U_{sui} = (s_i \times t_i)/c_i$ where c_i is the corresponding price SU_i is asked to pay for its spectrum utilization. Similarly, for a primary user $PU_j \in PU$, the Petri net $N_j = \{P_j, T_j, I_j, O_j, M_j\}$ for the price c_j and for its empty spectrum portion s_j, the movement from initial marking M_{opj} to final marking M_{fpj}, results in the maximization of its utility function U_{puj} such that $U_{puj} = c_j/(s_j \times t_j)$.

Definition 2. A Petri net $N_s = \{P_s, T_s, I_s, O_s, M_s\}$ for an empty part of the spectrum s, is said to be unsuccessful if moving from M_{os} to M_{fs}, results in no change in the utility functions of both the participating secondary and primary users such that:

$$U_{sui} = 0 \text{ and } U_{puj} = 0, \; \forall \{SU_i, PU_j\} \in \{SU, PU\} \quad \text{where } \{U_{sui}, U_{puj}\} \in \mathbb{R} \tag{2}$$

Definition 3. Let M_d be a desired markings set. The final marking set M_f is acceptable if and only if it represents the combination of both M_o and M_d i.e.

Table 1 Various Spectrum sharing states

p_{1}	SU5: ready to send CfP
p_{2}	PU2: PU agent’s cache (CfP arrives)
p_{3}	PU3: PU agent’s cache (CfP arrives)
p_{4}	PU2 : ready to receive CfP
p_{5}	PU3 : ready to receive CfP
p_{6}	SU5: CfP sent and wait for proposal(s)
p_{7}	PU2: CfP received
p_{8}	PU3: CfP received
p_{9}	SU3: SU agent’s cache (proposals arrive)
p_{10}	PU2: proposal sent and wait for the final response
p_{11}	PU3: proposal sent and wait for the final response
p_{12}	SU3: proposal received
p_{13}	PU2: PU agent’s cache (accept arrives)
p_{14}	PU3: PU agent’s cache (reject arrives)
p_{15}	PU2: temporary waiting phase
p_{16}	PU2: further CfP receiving stopped
p_{17}	PU3: reject received and temporary waiting phase
p_{18}	PU3: further CfP receiving stopped
p_{19}	PU2: ready to share the requested spectrum
p_{20}	SU5: ready to utilize spectrum
p_{21}	SU5: spectrum utilized and ready to pay price
p_{22}	PU2: spectrum shared and ready to receive price
p_{23}	SU5: price paid
p_{24}	PU2: price received

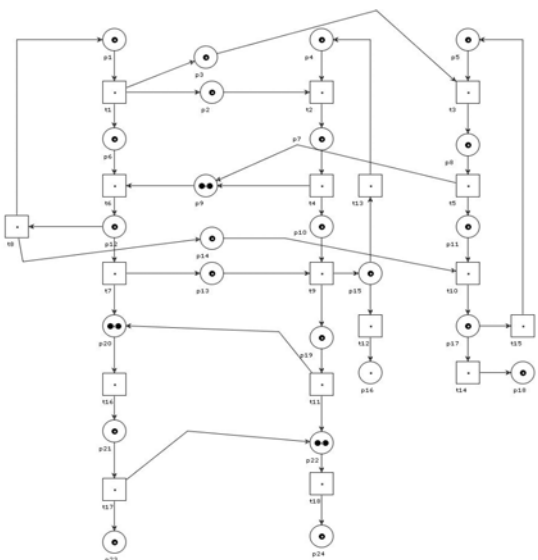


Fig. 2 An SU handling several proposals

$$M_f = M_o + M_d \quad (3)$$

From table 2, it is clear that, compared to M_o, there are one more token in places P_{16}, P_{18}, P_{20}, P_{22}, P_{23} and P_{24}. Thus, we can say that our PN-based model is not restricted to the initial marking M_o, since it is possible to increase the marking of at least one of the places to infinity.

Definition 4. a). The proposed PN is said to be *k*-bounded if the number of tokens in its places do not exceed *k* such that *k* is any integer > 0.
b). A PN is bounded when it is k-bounded.

Table 2 Several initial and final markings

	Mo	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	M_f
P1	1	-1							+1											1
P2	1	+1	-1																	1
P3	1	+1		-1																1
P4	1		-1											+1						1
P5	1			-1												+1				1
P6	1	+1					-1													1
P7	1		+1		-1															1
P8	1			+1		-1														1
P9	2				+1	-1	-2													2
P10	1				+1					-1										1
P11	1					+1					-1									1
P12	1						+1	-1	-1											0
P13	1							+1		-1										1
P14	1								+1		-1									1
P15	1									+1			-1	-1						0
P16	1													+1						2
P17	1										+1				-1	-1				0
P18	1														+1					2
P19	1									+1		-1								1
P20	2							+1				+1					-1			3
P21	1																+1	-1		1
P22	2											+1						+1	-1	3
P23	1																	+1		2
P24	1																		+1	2

The definition 4 is important in a sense that it is sometimes desirable to know whether the number of tokens in a cooperative PN is bounded in order either to define the size of the network or to check it for internal errors. For instance, some tokens may permanently stay in the places creating bottleneck situation, which is a serious drawback for cooperative networks.

5 Some Experimental Results

To conduct our experiments in JADE, we consider an ad-hoc scenario with multiple primary and secondary user agents and evaluate the performance of our PN-based cooperative approach. We randomly place a number of primary and secondary users in a specified area where each of the devices contains an agent deployed over it for cooperation purposes. For simplicity, two different fixed values of times (such as T1 and T2) are assumed, where Time 1 (T1) represents the short-term case and Time 2 (T2) is the longer period. When T1 is considered the SU agents can ask for the amount of spectrum within one hour limit (i.e.$0 \leq T1 \leq 60$Minutes) and similarly this limit is within two hours as in case of T2 (i.e.$0 \leq T2 \leq 120$Minutes). Our simulation starts with the total number of 6 SUs and 4 PUs, and for each next round there is an addition of 10 agents (i.e. 6 SUs and 4 PUs). The simulation is conducted for 10 subsequent rounds, with a total of 20 hours per day, for both T1 and T2 respectively and the average values of parameters are taken to draw the graphs. The PU agent's utility is represented as the price paid by SU agents for spectrum utilization divided by the amount of spectrum it has shared for the respected time period (holding time) as required by the SUs. The SU agent's utility is represented as its spectrum usage for the required time divided by the corresponding price paid to the PUs.

In Figure 3, we compare the average utility of primary and secondary user at T1 with those at T2 for different numbers of users (10, 20, 30,….). The figure shows that when the time limit is T2, the utilities are a bit less compared to the results obtained at T1. This is because the environment is partially mobile and some of the users are slightly hesitant to share their spectrum for longer periods. We observe that the average utility values show the linear behavior even with the increased number of agents.

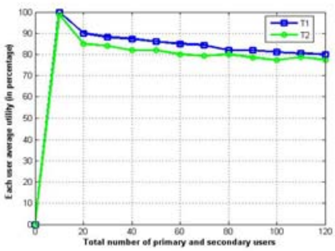


Fig. 3 Average utility of each agent

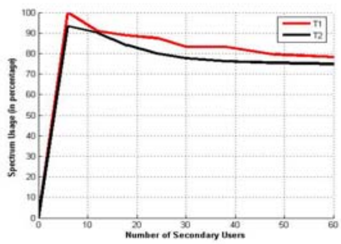


Fig. 4 Total spectrum usage

Figure 4 illustrates the percentage resource utilization of SUs over time periods T1 and T2. The beginning of the graph indicates that all the required resources are completely shared; whereas when the number of SUs reaches the middle values (30 to 40), approximately 90% of resources are shared. This spectrum sharing trend continues following the same pattern reaching higher values with achieved percentage of resources is between 75 to 85%. Thus, the performance degradation in terms of spectrum sharing is not high, even with large resource requirements.

6 Conclusion and Future Perspectives

In this paper, a model for dynamic spectrum allocations using cooperative multiagent system is presented. By detailing the spectrum allocation process using Petri net, we show that the approach is simple and effective in its working. While taking into account static spectrum allocations, we described a model for distributed approaches to allocate spectrum dynamically. The experimental results show that the proposed approach can absorb the high spectrum sharing demands by introducing the agents' cooperation between primary and secondary user devices. By examining the experimental results, it seems that high-quality utility based results can be achieved without a global control but with a cooperative solution in which agents rely on information from their neighborhoods. We are currently working on SU-to-SU cooperation based scenarios, in order to avoid unnecessary collisions and interferences. Our future work extends towards the comparison of our results with game-theoretical approaches.

References

1. Aydin, A., Altug, I.: Decentralized structural controller design for large scale discrete event systems modeled by Petri nets. Kybernetika, 3–14 (2009) ISSN 0023-5954
2. FCC. Spectrum policy task force report, ET Docket No. 02-135 (2002)
3. Hosseinabadi, G., Manshaei, H., Hubaux, J.-P.: Spectrum sharing games of infrastructure-based cognitive radio networks, Technical report, France (2008)
4. Hsieh, F.-S.: Developing cooperation mechanism for multi-agent systems with Petri nets. Engineering Applications of AI Journal 22, 616–627 (2009)
5. Jiang, X., Ivan, H., Anita, R.: Cognitive radio resource management using multi-agent systems. In: Consumer Communications and Networking Conference (CCNC 2007), pp. 1123–1127 (2007)
6. Kloeck, C., Jaekel, H., Jondra, F.: Multi-agent radio resource allocation. In: ACM Mobile Networks and Applications (MONET 2006), pp. 813–824 (2006)
7. Mir, U., Merghem-Boulahia, L., Gaïti, D.: Utilization of a cooperative multiagent system in the context of cognitive radio networks. In: International Week on Management of Networks and Services (MANWEEK 2009), pp. 100–104 (2009)
8. Mitola, J.: Cognitive radio: an integrated agent architecture for software defined radio, Ph.D Thesis, KTH Royal Institute of Technology, Sweden (2000)
9. Niyato, D., Hossain, E.: Competitive pricing for spectrum sharing in cognitive radio networks: dynamic game, inefficiency of Nash equilibrium, and collusion. IEEE Journal on Selected Areas in Communications 26, 192–202 (2008)
10. Panait, L., Luke, S.: Cooperative multi-agent systems learning: state of the art. In: Autonomous Agents and MultiAgent Systems (AAMAS 2005), pp. 387–434 (2005)
11. Weiss, G.: A modern approach to distributed artificial intelligence. MIT Press, USA (1999/2000)
12. Wellens, M., Riihijärvi, J., Mähönen, P.: Modelling primary system activity in dynamic spectrum access networks by aggregated on/off-processes. In: IEEE Workshop on Networking Technologies for SDR Networks, SECON 2009 (2009)
13. Zhang, C., Lesser, V., Shenoy, P.A.: Multi-agent learning approach to online distributed resource allocation. In: International Joint Conference on Artificial Intelligence (IJCAI 2009), pp. 361–366 (2009)

A Multi Agent Architecture for Tourism Recommendation

Laura Sebastia, Adriana Giret, and Inma Garcia

Abstract. In this paper, we present a Multi Agent System aimed to support an user on the realization of different leisure and tourist activities in a city. The system integrates agents that cooperate to dynamically capture the user profile and to obtain a list of suitable and satisfactory activities for the user, by using the experience acquired through the interaction of the user and similar users with the system. Moreover, the system is also able to generate a time schedule of the list of recommended activities thus forming a real activity plan. This paper focuses on the architecture and functional behaviour of our system.

1 Introduction

Nowadays, most people who plan a trip or a day-out will first initiate a search through the Internet. More and more people realize the advantages of the new technologies for planning leisure activities as an increasing number of companies and institutions offer tourist information which is easily accessible through web services. However, these sites provide the same piece of information to all users without considering the specific profile or needs of the user, which causes that the user has to process a large amount of information in order to pick up the preferable items.

e-Tourism [7] is an user-adapted tourism and leisure application. It recommends the user a list of activities to perform in the city of Valencia (Spain) by analyzing the captured user profile. It also computes a time schedule for the list of recommended activities taking into account the distance between places, the opening hours, etc. - that is an agenda of activities. We already have an implementation of this application, but we noticed some requirements that make it more appropriate to develop this application as a Multi-Agent System (MAS). These requirements include: new users should be able to enter the system at any time, existing users should be able

Laura Sebastia · Adriana Giret · Inma Garcia
Universidad Politecnica de Valencia, Spain
e-mail: {lstarin,agiret,ingarcia}@dsic.upv.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 547–554.
springerlink.com

to leave the system as well, tourism activities and information need to be updated accordingly, and new recommendation and planning techniques should be easily integrated. Moreover, cooperation scenarios should be created on demand depending on the tourism preferences of the user and the recommendation provided. To this end, the *e-Tourism* application requires a flexible architecture to implement multiple users, multiple general and particular tourism preferences, a negotiated planned agenda, different planning and recommendation techniques, etc. In this work we present a Multi-Agent architecture of *e-Tourism* focusing on the system components and its functional behaviour. This MAS architecture provides flexibility, openness, adaptability, and scalability to a tourism recommender and planning system.

This paper is organized as follows. Section 2 gives a brief state of the art of Multi-Agent RS. Section 3 introduces the MAS architecture of e-Tourism and describes the knowledge representation used. Sections 4 to 6 detail each agent and organization in this architecture. We finish with some conclusions and further work.

2 Background

A **recommender system** (RS) [6] is a personalization tool that attempts to provide people with lists of information items that best fit their individual tastes. A RS infers the user's preferences by analyzing the available user data, information about other users and information about the environment. Most of the implemented RS are either stand-alone or client-server systems [5]. Stand-alone systems can only infer the recommendations by inspecting some existing information source. There is no possibility for explicitly stating opinions and no possibilities of exchanging information between agents [2]. A Multi-Agent RS uses agents to help in the solution process, trying to improve the recommendation quality. The agents cooperate and negotiate in order to satisfy the users, interacting among themselves to complement their partial solutions or even to solve conflicts that may arise.

In the tourism domain, [3] proposes a MAS recommender approach where there is a community of agents in charge of searching information on the components of the travel package. The agents work in a distributed and cooperative way, sharing and negotiating knowledge with the global objective of recommending the best travel package to the user. *CASIS* [4] uses a metaphor from swarm intelligence to define a case-based RS with the global objective of recommending the best travel package to the user. Nevertheless, unlike *e-Tourism*, in these two approaches there is no record about previous users so that a new recommendation process to the same user must re-start from scratch. Besides these systems, mainly focused on recommending a travel package, in the tourism domain we can find other systems aimed at recommending a list of activities that a tourist can perform in a city. In addition, it is interesting to have the recommended visits organized as an agenda in order to visit as many places as possible and to minimize the trips between places. The definition of a tourist plan is a time consuming task that involves managing different kinds of information as opening hours, distances between places or the time spent on visiting each place. So, the task of the tourism recommender system is not only to help select the places to visit but also to organize a visit plan. In this paper we work on recommending tourist plans based on the user preferences and minimizing the trips between places. Our work is similar to *BerlinTainment* [10], which is a MAS that can be used to plan comprehensive day itineraries for entertainment on Berlin and determine current locations, points of interest, and routes. It provides the user with personalized recommendations (using a feature-based filtering) and location-based information. Our approach focuses on recommending a list of the activities that a tourist can perform in the city of Valencia. It also considers activities timetables and distances between the activities to compute the leisure and tourist agenda.

3 *e-Tourism* MAS Architecture Overview

The Multi-Agent architecture designed and built for the *e-Tourism* application has six main cooperation scenarios in which different agent types (roles) interact to achieve some goals. Figure 1 shows the different scenarios and the roles in the MAS architecture. We define four roles: the *User* role, to represent users; the *GRSK (Generalist Recommender System Kernel)* role, to represent the recommender system, the *Planner* role, to represent the planning system and the *Finder* role, to represent the information update mechanism. These roles are in charge of six use cases: (i) **Register User**: When a user first enters the system, the first step is to register and enter his personal details and general preferences. (ii) **Request Visit**: Each time the user enters the system for a new visit he will be requested to introduce his specific preferences for the current visit: the dates of the visit, his time schedule, if he is on his own or with children, etc. (iii) **Recommend Activities**: When a user requests a visit, the GRSK is in charge of generating a list of activities that are likely of interest to the user. (iv) **Plan Tourist Agenda**: From the list of recommended activities, the user selects those he is really interested in and discards those he does not want to be included in the final plan. At this stage the planning system schedules the activities according to the time restrictions of the user and the environment. (v) **Update User Profile**: When the user logs again in the system, he is asked to rate the activities in the last recommended plan. These ratings are used to improve the user profile and to gain more suitable recommendations. (vi) **Update Tourism Info**: The Finder role is in charge of keeping updated tourism information and activities in the system. In this way new activities can be added, existing ones can be deleted or modified, etc.

e-Tourism uses a **tourism ontology** to describe the features of the activities to recommend and the user profiles. It comprises a set of features that are commonly managed in a tourism domain like architectonic styles or types of buildings, extracted from the ”Art & Architecture Thesaurus”[1]. Unlike other ontologies (such as Harmonize[2]), which mainly cover accommodation facilities, events and activities, we need to represent feature of the locations and points of interest. Figure 2 shows part of the tourism ontology. An item i in *e-Tourism* (that is, an activity) is defined by a list of preferences.

[1] www.getty.edu/research/conducting-research/vocabularies/aat
[2] www.harmonise.org

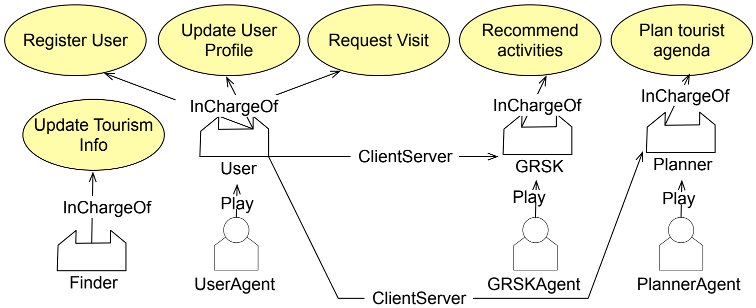


Fig. 1 Organization diagram and Use Cases of the e-Tourism system

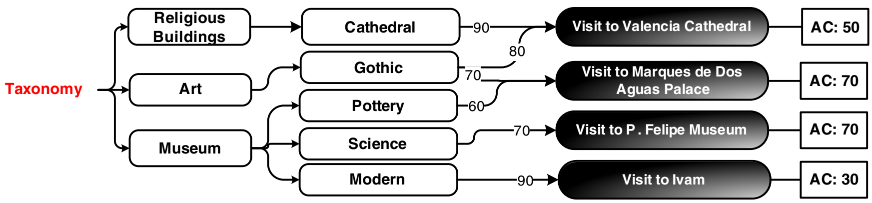


Fig. 2 Part of the *e-Tourism* ontology

Definition 1. *Let T be the set of all features in the ontology. A* preference *is a pair on the form (f_n^i, r_n^i), where $f_n^i \in T$ is a feature defined in the ontology; $r_n^i \in [0,100]$ is the degree of interest of f_n to the item i.*

For example, according to the ontology in figure 2, the description of "Visit to Valencia Cathedral" is set to $\{(\text{Gothic}, 80), (\text{Cathedral}, 90), \ldots\}$. The item "Visit to Marques de Dos Aguas Palace" is described by the pair (Gothic, 70) because it is considered to be a less interesting gothic building than the Cathedral. Additionally, activities are associated a value AC^i to represent how many times it has been performed when recommended. The information about items is updated by means of agents that play the Finder role.

4 UserAgent

The User role of figure 1 is played and implemented by one or more UserAgents. This agent represents a user of *e-Tourism* that wants to plan a visit.

A UserAgent (figure 3(a)) stores a **user profile** that includes personal and demographic details about the user, like age, gender, family or country. This profile can be initialized, modified or consulted by means of the *Set*, *Change* and *Get Profile* tasks, respectively. These tasks are executed whenever the *Register User* and *Update User Profile* Use Cases of figure 1 are executed. Second, the user general preferences model, denoted by GP^u (**General Preferences**), contains the description of

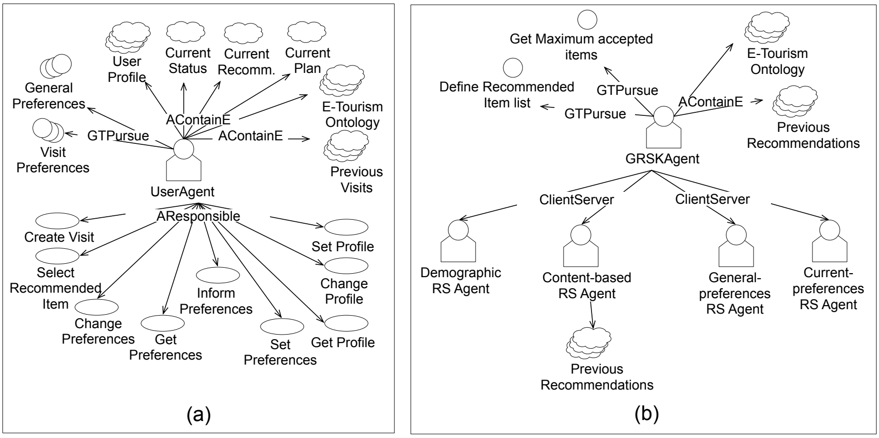


Fig. 3 (a) Agent diagram of the UserAgent and (b) Agents of the GRSK organization

the types of items the user *u* is interested in. This description is composed by a set of preferences (definition 1). We consider GP^u as a goal for our user agent (figure 3(a)) because these preferences must be fulfilled by the recommendation provided to the user. These preferences are initialized when the *Register User* Use Case is executed and handled by means of the *Set*, *Change*, *Inform* and *Get Preferences* tasks.

Each time the user enters the system for a new visit (*CreateVisit* task), that is whenever the *Request Visit* Use Case is executed, he will be requested to introduce his specific preferences for the current visit (**Visit Preferences**), which may differ from his general preferences. For example, unlike other user trips, he might not be traveling with children in the current visit. The visit specific preferences are also defined by a set of preferences (definition 1). Moreover, the user also indicates his current location, which is stored in the **Current Status** and the maximum number *N* of recommendations he desires. The UserAgent keeps data about the current interaction of the user with the system. The list of recommended items provided by the system to the UserAgent (as a result of the *Recommend Activities* Use Case of figure 1) is stored as the **Current Recommendation**. From this list of recommendations, the user picks up the activities he is really interested in, and discards those ones he does not want to be included in the final agenda. The remaining items are marked as indifferent. This process is performed by means of the *SelectRecommendedItem* task. The list of selected and indifferent items is considered to build the agenda of activities, that is, the **Current Plan** in the *Plan Tourism Agenda* Use Case of figure 1 with the help of the PlannerAgent (see section 6). The result of the planning process is a list of activities joint with an specific start time and a duration.

When the user is recommended an activity, he can perform this activity or not. Moreover, if the user performs the activity, it can be satisfactory (at a certain degree) or not. This information is crucial for the system because it can be considered in order to improve future recommendations. Then, when the user logs again in the system for a new query, he is asked to rate the items recommended in the previous interaction. The user will specify which activities he has performed and the degree of satisfaction executing the *Update User Profile* Use Case of figure 1. This information is stored as **Previous Visits**.

5 GRSK Organization

The main goal of the *Generalist Recommender System Kernel (GRSK)* is to generate the list of activities to recommend to the user. The GRSK is generalist in the sense that it can work with any application domain as long as the data of the new domain can be defined through an ontology representation.

The core of the GRSK (figure 3(b)) is called the **GRSKAgent**, which is the manager of the recommender organization. In order to respond to a request for a recommendation by a UserAgent, it invokes the different RS agents which derive a set of positive and negative constraints: CP^u, the positive constraints, which denote the preferences (definition 1) that the recommended items must meet; and CN^u, the negative constraints, which denote the preferences that the recommended items must not fulfill. The items that match these constraints are recommended. The GRSK has been designed to have the possibility of easily adding new recommendation techniques. This is so thanks to the distributed GRSK architecture in which every recommendation technique is encapsulated into an agent: this way a new technique can be easily added by means of a new agent compliant with the interaction protocol detailed below. The functioning of the four basic RS techniques are explained in [7].

The process to obtain a recommendation is performed as follows. A user is described by a user profile and a set of general and specific preferences which are used by the RS to infer the set of positive and negative constraints. Each RS agent returns a set of CP^u except the current-preferences-based filtering agent, which returns both a set CP^u and a set CN^u. These RS agents can obtain a different set of constraints or a different degree of interest for the same feature. Moreover, they are autonomous to decide whether the constraints they have obtained are accurate enough to be considered to obtain the final recommendation. Once all the RS have computed the corresponding CP^u and CN^u, the second step in the recommendation process is executed. The first task is to obtain a joint list of constraints from the lists returned by each RS agent ($Mix(CP, CN)$). This is computed as follows. For each feature f_n included in a list of positive constraints CP^u returned by a RS agent, a pair (f_n, r_n) is added to the final list of positive constraints CP', where r_n is the average of the values associated to f_n in all the positive constraints lists. The same process is applied to obtain the final list of negative constraints CN'. Then, the list of items that match CP' and CN' is computed ($GetItems(CP', CN')$). An item i matches CP' and CN' if it satisfies a positive constraint, it does not satisfy any negative constraint and it has not already been accepted by the user. The GRSKAgent computes a priority for each item in this list according to the values of AC^i, the degree of interest associated in the ontology and the estimated degree of interest calculated by the RS agents. The list of items is ordered according to the computed priority. The GRSKAgent selects the N best recommendations, which is the set of recommended items to the user u. This list

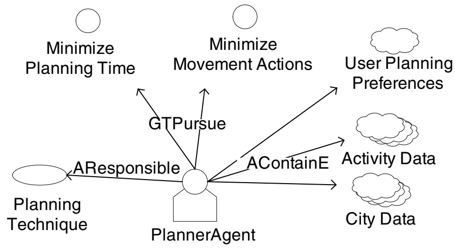


Fig. 4 Agent diagram of the PlannerAgent

is finally recommended to the UserAgent. At this moment, the UserAgent executes the *SelectRecommendedItems* task to obtain the filtered recommended items, which will be used by the PlannerAgent to build the agenda.

6 PlannerAgent

The PlannerAgent (figure 4) is responsible of computing a feasible plan from the list of activities recommended by the GRSKAgent and then filtered by the user. It manages three groups of different data: (a) the **User Planning Preferences**, such as the visit date, the user available time, the current geographical location of the user, etc.; (b) the information about each activity (**Activity Data**) such as the opening hours of each activity, the address of the place where the activity takes place, the duration of activity, etc.; (c) information about the city map (**City Data**), which comprises the city streets and the streets intersections.

All these data are properly analyzed and combined by the PlannerAgent to build the user-adapted planning problem. The PlannerAgent must select which activities among all the filtered activities to include in the plan, because not all these activities will be likely included in the plan since the scheduling will depend on the user available time, his temporal constraints and the time restrictions of the environment (i.e. opening hours of places). Therefore, we are facing a *Partial Satisfaction Planning (PSP)* problem [8]. Unlike classical planning problems, in PSP problems the solution plan is not required to achieve all the goals but instead achieve the *best* subset of goals given the resource limitations. In the particular context of the calculation of a tourist agenda for a given user, the goal is to obtain a plan with the most satisfactory activities (as possible), trying to minimize the time spent on going from one place to another. This process is detailed in [7].

7 Conclusions and Further Work

e-Tourism is a multi-agent system that generates personalized recommendations about tourist tours in the city of Valencia (Spain). It is intended to be a service for foreigners and locals to become deeply familiar with the city and plan leisure activities. *e-Tourism* makes recommendations based on the user's tastes, his demographic classification, the places visited by the user in former trips and, finally, his current visit preferences. The tool shows the user an agenda of recommended activities which reflect the user's tastes and takes into account the geographical distance between places and the opening hours of these places.

A future research line is to extend *e-Tourism* for group recommendation, calculating the list of activities according to the global or particular constraints rather than in terms of the group preferences. Our aim is to apply agreement technologies for group recommendation, in order to increase the reliability of electronic communities by introducing human social control mechanisms.

Acknowledgements. Partial support provided by Consolider Ingenio 2010 CSD2007-00022, Spanish Government Project MICINN TIN2008-6701-C03-01 and Valencian Government Project Prometeo 2008/051.

References

1. Birukov, A., Blanzieri, E., Giorgini, P.: Implicit: A recommender system that uses implicit knowledge to produce suggestions. In: Proceedings of Nineteen International Joint Conference on Artificial Intelligence IJCAI 2005, Workshop Multi-Agent Information Retrieval and Recommender Systems (2005)
2. Foner, L.: Yenta: A multi-agent, referral-based matchmaking system. In: Proceedings of ACM Autonomous Agents 1997, pp. 301–307 (1997)
3. Lorenzi, F., Bazzan, A., Abel, M.: An architecture for a multiagent recommender system in travel recommendation scenarios. In: ECAI workshop on Recommender Systems (2006)
4. Lorenzi, F., Santos, D.S., Bazzan, A.L.C.: Negotiation for task allocation among agents in case-base recommender systems: a swarm-intelligence approach. In: Proceedings of the Workshop Multi-Agent Information Retrieval and Recommender Systems - Nineteenth International Conference on Artificial Intelligence, IJCAI 2005 (2005)
5. Niinivaara, O.: Agent-based recommender systems. Tech. rep., Department of Computer Science, University of Helsinki (2004)
6. Resnick, P., Varian, H.: Recommender systems. Communications of the ACM 40(3) (1997)
7. Sebastia, L., Garcia, I., Onaindia, E., Guzman, C.: e-Tourism: a tourist recommendation and planning application. International Journal on Artificial Intelligence Tools (WSPC-IJAIT) 18(5), 717–738 (2009)
8. Smith, D.E.: Choosing objectives in over-subscription planning. In: Proceedings of the International Conference on Automated Planning and Scheduling (2004)
9. Wei, Y.Z., Jennings, N.R., Moreau, L., Hall, W.: User evaluation of a market-based recommender system. Journal of Autonomous Agents and Multi-Agent Systems 17(2), 251–269 (2008)
10. Wohltorf, J., Cissee, R., Rieger, A., Scheunemann, H.: Berlintainment: An agent-based serviceware framework for context-aware services. In: 1st International Symposium on Wireless Communication Systems (2004)

Improving Functionalities in a Multi-agent Architecture for Ocean Monitoring

Fernando de la Prieta, Oscar Gil, Carolina Zato,
Beatriz Martín, and Alberto García

Abstract. This paper presents an improved version of a multiagent architecture aimed at providing solutions for monitoring the interaction between the atmosphere and the ocean. The ocean surface and the atmosphere exchange carbon dioxide. This process is can be modeled by a multiagent system with advanced learning and adaption capabilities. The proposed multiagent architecture incorporates CBR-agents. The CBR-agents proposed in this paper integrate novel strategies that both monitor the parameters that affect the interaction, and facilitate the creation of models. The system was tested and this paper presents the results obtained.

Keywords: Multi-Agent Systems, Intelligent agents, Distributed Computing.

1 Introduction

Multi-agent systems are very appropriate for resolving problems in a distributed way [15]. Agents have a set of characteristics, such as autonomy, reasoning, reactivity, social abilities, pro-activity, mobility, organization, etc. which allow them to cover several needs for dynamic environments. Agent and multi-agent systems have been successfully applied to several scenarios, such as education, culture, entertainment, medicine, robotics, etc. [3], [21]. Moreover, the continuous advancement in mobile computing makes it possible to obtain information about the context and also to react physically to it in more innovative ways [15]. Nevertheless, complex systems need higher adaptation, learning and autonomy levels than pure BDI model [1]. This can be achieved by modelling the agents' characteristics [23] to provide them with mechanisms that allow solving complex problems and autonomous learning [8].

One of the factors of greatest concern in climactic behaviour is the quantity of carbon dioxide present in the atmosphere. Carbon dioxide is one of the greenhouse

Fernando de la Prieta · Oscar Gil · Carolina Zato · Beatriz Martín · Alberto García
Facultad de Ciencias
Universidad de Salamanca
Plaza de la Merced s/n, 37008, Salamanca, Spain
e-mail: {fer,oscar.gil,carol_zato,eureka,alberto_gar}@usal.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 555–562.
springerlink.com

gases that helps to make the earth's temperature habitable, so long as it is maintained at a certain level [20]. The main system regulating carbon dioxide in the atmosphere has traditionally been thought to be the photosynthesis and respiration of plants. However, teledetection techniques have been able to show that the ocean plays a highly important role in the regulation of carbon quantities, although the full significance of this still needs to be determined [12, 20]. Current technology allows us to obtain data and make calculations that were unthinkable some time ago. This data provides an insight into the original source of carbon dioxide, the decrease in carbon dioxide, and the causes for both [16], which allows us to make predictions about the behaviour of carbon dioxide in the future.

The aim of the present study is to improve the functionalities of an architecture that makes it possible to construct dynamic systems capable of growing in dimension and adapting their knowledge to environmental changes [3, 8]. The mission of the multiagent system is to globally monitor the interaction between the ocean surface and the atmosphere, facilitating the work of oceanographers. The system is being used in order to evaluate and predict the amount of carbon dioxide (CO_2) absorbed or expelled by the ocean in the North Atlantic [6]. The CBR-BDI agents [5] presented in the framework of this research incorporate innovative techniques in each of the stages of the CBR cycle. The retrieve phase incorporates a novel strategy based on growing cell structure neural network that provides a set of cases grouped in meshes according to similarity criteria. The reuse phase is composed of a multilayer perceptron neural network and a Jacobean sensitive matrix. The revise phase is carried out by means of a pondered weight technique. Finally, the retain stage updates the growing cell structure neural network.

The next section reviews the environmental problem that motivates the majority of this research. Section three describes the multiagent architecture specifically developed to monitor the air-sea interaction. Section four presents the CBR-BDI agent based system developed. Finally the conclusions and some preliminary results are presented.

2 Problem Description and Background

In recent years a great interest has emerged in climactic behaviour and the impact that mankind has had on the climate. One of the most worrying factors is the quantity of CO_2 present in the atmosphere. Until only a few years ago, the photosynthesis and breathing processes in plants were considered to be the regulatory system that controls the presence of CO_2 in the atmosphere. However, the role of the ocean in the regulation of carbon volume is very significant and so far remains indefinite [19]. Current technology makes it possible to obtain data and estimates that were beyond expectations only a few years ago. The goal of this project is to construct a model that calculates the global air-sea flux of CO_2 exchanged between the atmosphere and the surface waters of the ocean. In order to create a new model for the CO_2 exchange between the atmosphere and the oceanic surface a number of important parameters must be taken into consideration: sea surface temperature, air temperature, sea surface salinity, atmospheric and hydrostatic pressures, the presence of nutrients and the wind speed vector (module and direction) [20].

These parameters can be obtained from oceanographic ships as well as from satellite images. Satellites provide a great amount of daily information and there is a growing need for the ability to automatically process and learn from this source of knowledge. These parameters allow us to calculate the variables that define our models, such as the velocity of gas transfer, solubility, or the differentiation between partial pressures on the atmosphere and sea surface (a case structure is shown in Table 1).

Table 1 Case Attributes

Case Field	Measurement
DATE	Date (dd/mm/yyyy)
LAT	Latitude (decimal degrees)
LONG	Longitude (decimal degrees)
SST	Temperature (ºC)
S	Salinity (unitless)
WS	Wind strength (m/s)
WD	Wind direction (unitless)
Fluo_calibrated	fluorescence calibrated with chlorophyll
SW pCO_2	surface partial pressure of CO_2 (micro Atmospheres)
Air pCO_2	air partial pressure of CO_2 (micro Atmospheres)
Flux of CO_2	CO_2 exchange flux (Moles/m^2)

3 Multiagent System for Predicting the Ocean Behaviour

To handle all the potentially useful data to create daily models in a reasonable time and with a reasonable cost, it is necessary to use automated distributed systems capable of incorporating new knowledge. Our proposal consists of a multiagent system whose main characteristic is the use of CBR-BDI agents. The architecture was detailed in previous works [3, 8] and is shown in Figure 1.

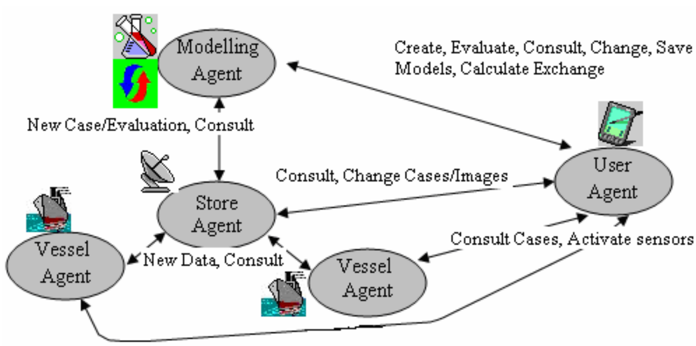


Fig. 1 Diagram of the architecture of our MAS

Figure 1 illustrates a multiagent system in which is it possible to observe how a Modelling agent with a CBR-BDI architecture is responsible for the creation and evaluation of models in terms of the data received from the Store, Vessel and User agents. This model makes it possible to monitor and predict the carbon dioxide exchange between the ocean surface and the atmosphere. The Store agent processes the images from the satellite and transforms them for use by the system. Each Vessel agent is installed in a ship and collects information in-situ that makes it possible to evaluate the models created by the Modelling agent. The User agent can interact with any of the other agents [22]. Figure 1 shows how the agents interact with each other and with their surroundings [1, 4]. In order to resolve the problem from an oceanographic perspective, the ocean was divided into a series of zones in each of which there is a Modelling Agent, a Store Agent, and various Vessel Agents.

This paper presents an improved version of the Modelling agent [3, 8], incorporating novel strategies in the retrieve, reuse and revise stages of the CBR-agent cycle. These innovative strategies are presented in detail in the following section.

4 Agents with Advanced Prediction Abilities

The deliberative agents proposed in the framework of this investigation use the concept of Case-based Reasoning (CBR), a type of reasoning based on the use of past experiences [14], to gain autonomy and improve their problem-solving capabilities. The method proposed in [5] facilitates the incorporation of case-based reasoning systems as a deliberative mechanism within BDI agents, allowing them to learn and adapt themselves, lending them a greater level of autonomy than pure BDI architectures [1]. Accordingly, CBR-agents implemented using case-based reasoning systems can reason autonomously and therefore adapt themselves to environmental changes. The case-based reasoning system is completely integrated within the agents' architecture.

The Modelling agent, a CBR-BDI agent that has two principal functions. The first function is to generate models that are capable of predicting the atmospheric/oceanic interaction in a particular area of the ocean in advance. The second is to permit the use of such models. The reasoning cycle of a CBR system is included among the activities, and is comprised of the retrieval, reuse, revise and retain stages. An additional stage is used to introduce an expert's knowledge. This reasoning cycle must correspond to the sequential execution of some of the agent roles. The Modelling agent carries out roles to generate models such as Jacobean Sensitivity Matrix (JSM), Pondered Weigh Technique (PWT), Revision Simulated Equation (RSE), and other roles that allow it to operate with the calculating models, like Forecast Exchange Rate, Evaluate Model or Consult model. The roles used to carry out the stages of the CBR cycle are described as follows.

The content of the information stored in the memory of cases for each of the cases is described in Table 1. As can be seen, a case consists of a series of variables that can be represented as a tuple c=(d, l, o, t, s, w, r, f, p, a, M, e, x), where d represents the date, l the latitude, o the longitude, t the temperature, s the salinity, w the wind strength, wd the wind direction, f the fluorescence calibrated, s the surface partial pressure of CO_2, a the air partial pressure of CO_2, M the Multilayer Perceptron (MLP) associated to the case, e the CO_2 Flux, and i the exchange value. The memory of cases is defined as a set of cases and is represented as C={c}. When a new problem is studied, the system incorporates a new case c_{n+1} and a new CBR cycle is executed:

Retrieve phase: The retrieval process identifies those cases in the memory of cases that have the highest level of similarity with the new case $c_{n+1.}$ In order to do so, the memory of cases is structured in such a way so as to group together the most similar cases. GCS [13] (Growing Cell Structure) is used at this stage, since GCS does not set the number of neurons, or the degree of connectivity. GCS networks adjust the data by means of a series of disconnected meshes that are obtained during the training stage of the neural network. In this sense, the neural network provides a series of distributed meshes that represent the memory of cases. Each of the cases of the memory of cases is assigned to the nearest mesh, so when a new case is studied, the closest mesh is selected along with the cases associated to the mesh. These are the cases that will be used in the reuse phase. If there exists is a neural network that has been previously trained with the set of retrieved cases, that is, a CBR cycle previously executed with the cases of the selected mesh, then the settings of the neural network are reloaded. When the training process finishes, the result obtained is a set of cases grouped in meshes that are represented as $G = \{g_i / g_i \subseteq C\}$, where $g_i \cap g_j = \phi \forall i \neq j$.

Reuse phase: This phase is carried out by means of a multilayer perceptron [18]. The MLP makes only use of the data recovered in the retrieve phase instead of working with all the data stored in the memory of cases. This fact provides a notable reduction in the time required for the training stage of the MLP, and improves the prediction provided by the neural network since the data are more homogeneous. When the group g_i has already executed a Reuse phase and, as a result, it is associated with a previously trained MLP, then it is necessary to calculate the estimate error rate for the cases used by the MLP. If the condition established by Equation (1) is met, the training stage is not carried out.

$$\frac{\sum_{i=1}^{N} | M_{g_j}(c_i) - x_i |}{N \cdot \bar{x}} < \mu \tag{1}$$

where N represents the set of cases for the group g_j, $M_{g_j}(c_i)$ is the value estimated by the MLP for the case c_i, x_i is the exchange value, and μ is the threshold that identifies the limit considered as valid. Otherwise, when a MLP does not previously exist, it is necessary to execute the training phase before making predictions. To carry out the training phase of the MLP, it is necessary to readjust the data in such a way that all the data are normalized in the interval [0.2-0.8]. In the input layer of the MLP there is a neuron for each of the parameters shown in Table 1, except the Flux of CO_2, which is the solution for the cases. The number of neurons selected for the hidden layer of the MLP is determined using the expression 2n+1, where n is the number of neurons in the input layer. This value was defined following the criteria proposed by Kolmogorov [18]. Finally, the output layer of the MLP is composed of a neuron that represents the Flux of CO_2 parameter shown in Table 1. The training stage finishes when the cross validation, which uses 10% of the initial cases, provides an error rate that is lower than μ . Once the MLP has been trained, the Jacobean Sensitivity Matrix is calculated and the Pondered Weigh Technique is applied.

Revise phase: Revision Simulated Equation (RSE): During the revision stage an equation (F) is used to validate the proposed solution p*.

$$F = kso(pCO_2SW - pCO_2AIR) \tag{2}$$

Where F is the flux of CO2, *k* is the gas transfer velocity (3), *so* is the solubility verifying (4) and pCO2 is the partial pressure of CO2 (5).

$$k = (-5{,}204\,Lat + 0{,}729\,Long + 2562{,}765)/3600 \tag{3}$$

$$so = e^{\left(\frac{93{,}4517}{100tk} - 60{,}2409 + 23{,}3585\log(100tk) + s(0{,}023517 - 0{,}023656 \bullet 100tk + 0{,}0047036 \bullet 1002tk)\right)} \tag{4}$$

$$pCO_2 = A + BLong + CLat + DSST + EYear \tag{5}$$

As can be seen in (6), k depends on Lat (Latitude), Long (Longitude). As can be seen in (7) *so* depends on tk = 273,15 + t. where t is the temperature and s is the salinity. Finally, in (8) it is possible to observe that pCO2 depends on the SST, which is the temperature of the marine surface or air as it corresponds to pCO2SW or pCO2AIR. The coefficients of the equation (8) depend on the month.

Retain phase: This phase begins once the prediction has been compared to the result provided using the mathematical model. If the case is considered valid (if the prediction differs by less than 10%), the case is stored in the memory of cases. When this occurs, it is necessary to train the GCS network in order to include the new case in the structure of the memory of cases. In this way the new experience obtained processing the current case will be taken into consideration for the next prediction.

5 Results and Conclusions

The system described above was tested in the North Atlantic Ocean during 2003 and 2004. Although the system is not fully operational and the aim of the project is to construct a research prototype and not a commercial tool, the initial results have been very successful from a technical and scientific point of view. The construction of the distributed system was relatively simple, using previously developed CBR-BDI libraries [3, 4, 5, 3, 6]. facilitates the straight mapping between the agent definition and the CBR construction. The multiagent system automatically incorporated over 50,000 instances during the five months and eliminated 12% of the initial ones.

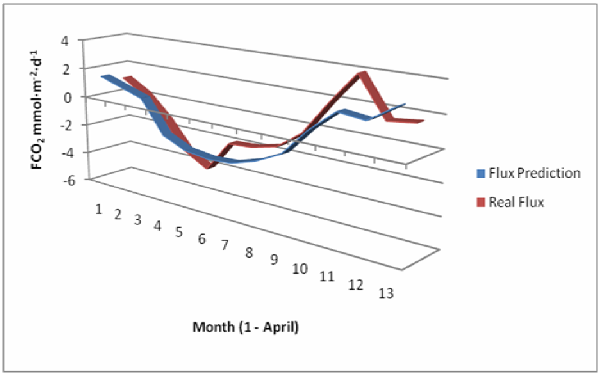


Fig. 2 Real CO2 flux and flux prediction

Figure 2 shows a comparison of the real data and the predictions provided by the multiagent system working with data from 2003-2004. As shown in Figure 2 the predictions provided by the multiagent system are accurate (9 of the 12 models were accepted as successful). The multiagent system makes predictions based on previous experiences, taking into account the similitude with past situations. Figure 2 shows how the precision of the prediction improves when the number of cases increases.

On the other hand, it is necessary to control the number of cases in the memory of cases in order to avoid an excessive growth of the cases available. To maintain the memory of cases, we used a strategy based on priorities, consisting of a pyramidal structure of efficiencies. The multiagent system facilitates the incorporation of new agents that use different modeling techniques and learning strategies, so our future work will focus on the incorporation of new agents with alternative techniques and the execution of additional experiments.

Acknowledgements. This work has been partially supported by the JCYL SA071A08 project.

References

1. Andrade, F., Novais, P., Machado, J., Neves, J.: Contracting agents: legal personality and representation. Artif. Intell. Law 15(4), 357–373 (2007)
2. Bratman, M.E., Israel, D., Pollack, M.E.: Plans and resource-bounded practical reasoning. Computational Intelligence 4, 349–355 (1988)
3. Bajo, J., Corchado, J.M., Alonso, V.: Sistema Multiagente para la Predicción del Intercambio de CO_2 en el Océano Atlántico Norte. IEEE Latin America Transactions 6(6), 505–510 (2008)
4. Chaib-draa, B., Dignum, F.: Trends in Agent Communication Language. Computational Intelligence 18(2), 89–101 (2002)
5. Corchado, J.M., Laza, R.: Constructing Deliberative Agents with Case-based Reasoning Technology. International Journal of Intelligent Systems 18(12), 1227–1241 (2003)

6. Corchado, J.M., Aiken, J., Corchado, E., Lefevre, N., Smyth, T.: Quantifying the Ocean's CO2 Budget with a CoHeL-IBR System. In: Funk, P., González Calero, P.A. (eds.) ECCBR 2004. LNCS (LNAI), vol. 3155, pp. 533–546. Springer, Heidelberg (2004)
7. Corchado, J.M., Bajo, J., Abraham, A.: GERAmI: Improving the delivery of health care. IEEE Intelligent Systems 23(2), 19–25 (2008)
8. Corchado, J.M., Aiken, J., Bajo, J.: A CBP Agent for Monitoring the CO2 Exchange Rate. Studies on Computational Intelligence, vol. 73, pp. 213–246 (2008)
9. Corchado, J.M., Bajo, J., de Paz, J.F., Rodríguez, S.: An execution time neural-CBR guidance system. NeuroComputing 72(13-15), 2743–2753, ISSN 0925-2312; Special Issue on Hybrid Intelligence in Neurocomputing (2009), doi:10.1016/j.neucom.2008.08.020
10. Corchado, J.M., Gonzalez-Bedia, M., De Paz, Y., Bajo, J., De Paz, J.F.: Replanning mechanism for deliberative agents in dynamic changing environments. Computational Intelligence 24(2), 77–107 (2008)
11. Dransfeld, S., Tatnall, A.R., Robinson, I.S., Mobley, C.D.: A comparison of Multilayer Perceptron and multilinear regression algorithms for the inversion of synthetic ocean colour spectra. Int. J. Remote Sens. 25(21), 4829–4834 (2004)
12. Ella Hassanien, A., Abraham, A., Grosan, C.: Spiking neural network and wavelets for hiding iris data in digital images. Soft Computing 13(4), 401–416 (2009)
13. Fritzke, B.: A growing neural gas network learns topologies. In: Tesauro, G., Touretzky, D.S., Leen, T.K. (eds.) Advances in Neural Information Processing Systems, Cambridge, MA, vol. 7, pp. 625–632 (1995)
14. Hruschka, E.R., Campello, R., Freitas, A., Carvalho, A.: A Survey of Evolutionary Algorithms for Clustering. IEEE Transactions on Systems, Man and Cybernetics - Part C: Applications and Reviews 39(2), 133–155 (2009)
15. Jayaputera, G.T., Zaslavsky, A.B., Loke, S.W.: Enabling run-time composition and support for heterogeneous pervasive multi-agent systems. Journal of Systems and Software 80(12), 2039–2062 (2007)
16. Lefevre, N., Aiken, J., Rutllant, J., Daneri, G., Lavender, S., Smyth, T.: Observations of pCO2 in the coastal upwelling off Chile: Sapatial and temporal extrapolation using satellite data. Journal of Geophysical Research 107 (2002)
17. Martinetz, T.: Competitive Hebbian learning rule forms perfectly topology preserving maps. In: Gielen, S., Kappen, B. (eds.) ICANN 1993: International Conference on Artificial Neural Networks, pp. 427–434. Springer, Amsterdam (1991)
18. Murtagh, F.: Multilayer perceptronsnext term for classification and regression. Neurocomputing 2(5-6), 183–197 (1991)
19. Saitou, N., Nie, M.: The neighbor-joining method "A new method for reconstructing phylogenetic trees". Mol. Biol. 4, 406–425 (1987)
20. Sarmiento, J.L., Dender, M.: Carbon biogeochemistry and climate change. Photosynthesis Research 39, 209–234 (1994)
21. Schön, B., O'Hare, G.M.P., Duffy, B.R., Martin, A.N., Bradley, J.F.: Agent Assistance for 3D World Navigation. In: Panayiotopoulos, T., Gratch, J., Aylett, R.S., Ballin, D., Olivier, P., Rist, T. (eds.) IVA 2005. LNCS (LNAI), vol. 3661, p. 499. Springer, Heidelberg (2005)
22. Sheng, Y., Chen, G., Tan, K., Deshpande, U., Vance, B., Ho, Y., McDonald, C., Henderson, T., Kotz, D., Campbell, A., Wright, J.: MAP: A scalable monitoring system for dependable 802.11 wireless networks. IEEE Wireless Communications 15(5), 10–18 (2008)
23. Wooldridge, M., Jennings, N.R.: Intelligent Agents: Theory and Practice. The Knowledge Engineering Review 10(2), 115–152 (1995)

Multiagent Systems in Stock Index Prediction

Ricardo Antonello and Ricardo Azambuja Silveira

Abstract. This paper describes a method to reduce the uncertainty about investments in stock market. The objective is to predict the movements of the Bovespa Stock Index through multiagent system. The proposed system has a population of agents for each asset that is a constituent part of the stock index. Agents have learning skills due to the use of neural networks. Predict the direction of the populations of agents is more efficient than predict the direction based only on data in the index time series.

Keywords: Multi-agent systems, time-series prediction, neural networks.

1 Introduction

In finance world the hypothesis known as the Efficient Market Hypothesis (EMH) implies that all the financial agents are rational and have at the same time, the same set of information available, in other words, information and expectations are reflected correctly and immediately in the price of assets. (Fama 1970; 1991).

According to this theory, the fluctuation of the price of shares and financial assets is randomic, and the best possible forecast for tomorrow is the price of today and does not exist distortions in the prices of assets, since prices reflect all variables available. Thus, there is no way to protect the investor forecasting the future movements of the market and no investor would be able to obtain yields above the average market or to protect its assets applied to possible financial crises. However, there is not consensus on this theory (Malkiel 1999) and several empirical initiatives have shown that the claim of EMH is not entirely correct (Farmer 1996).

There are basically two main research areas that, contrary to the EMH, try to forecast the future direction of movements. The first is an econometric point of

Ricardo Antonello · Ricardo Azambuja Silveira
Statistic and Informatics Department
Federal University of Santa Catarina State (UFSC)
Florianópolis, SC, Brazil
e-mail: ricardo@antonello.com.br, silveira@inf.ufsc.br

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 563–571.
springerlink.com

view and the second major area of research deals with the theory of modeling Multiagent Market (Theory of Multi-Agent Market Modeling) that focuses on micro-structure of the market on the assumption that the movements of prices emerge from the interaction of many individual actors of the market. (Bar-Yan 1997 apud Grothmann 2002).

The modeling of financial markets and other social phenomena based on multi-agent systems have attracted the attention of many researchers in recent years (Gupta 2001) (Grothmann 2002) (Araújo 2004) (Galla 2006). Several phenomena can be studied with this type of modeling including studies of growth of epidemics, traffic of vehicles or pedestrians and investors in the stock exchange.

The multiagent model markets are an alternative to Efficient Market Hypothesis (EMH) (Fama 1970; 1991) or econometric models such as linear regression or neural networks, because it close much of the actual dynamics of financial markets as they are able to "capture the complexity and dynamics of behaviour and learning in financial markets using models of markets, strategies and more realistic structures" (Farmer 1996 apud Grothmann 2002).

This paper describes the use of a communities' behaviour of software agents with cognitive capacity to forecast the direction of Bovespa Index, the main index of the Bolsa de Valores de São Paulo – Bovespa. Bovespa is the main Stock Market in Brazil. Each asset in the index has an agent population that will be considered as the weight of the asset in the index to make the final decision to predict if the Bovespa Index rises or falls in the following period.

2 The Bovespa Index

Index of stock portfolios are a set of most representative stock on the market they represent. There are several indices that reflect the evolution of various international markets, among which we mention the Dow Jones, SP500 and Nasdaq in the USA, the CAC of France, the UK's FTSE, the DAX of Germany, the Nikkei of Japan and the Hang Seng Hong Kong.

The Bovespa is the main stock market in Brazil, and the Bovespa Index (Ibov) is the most important performance indicator of the average price of Brazilian stock market. The Ibovespa (Ibov), shows the behavior of the main stocks traded at Bovespa. The index maintained the integrity of its historical series without any methodological change since its implementation in 1968.

Currently there are 66 shares in the index (portfolio of January to April 2009) (Bovespa 2009), which are selected based on criteria of liquidity, in other words, the most traded, the greater the chances of share being in the index. The Ibovespa is recalculated every four months and takes into account the movement of the last twelve months. In addition, each stock requires participation in terms of volume, more than 0.1% of the total period, and was negotiated by more than 80% of the total stock of the period.

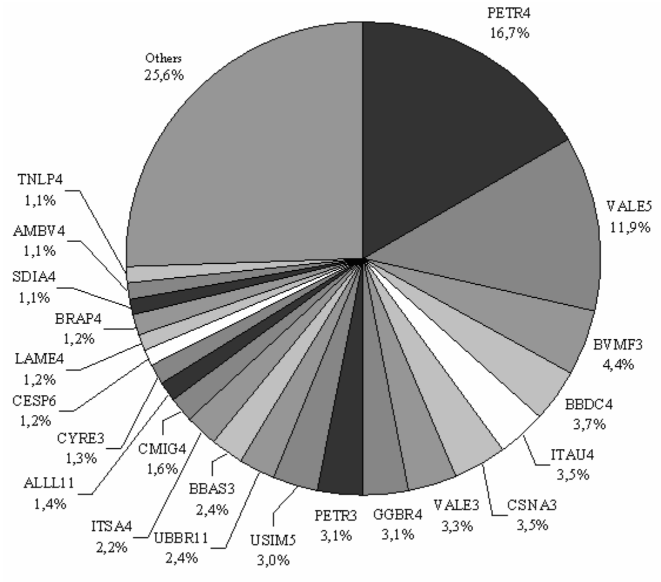


Fig. 1 Shares that are part of the Bovespa Index with their respective weights (Bovespa 2009)

3 Multiagent Systems for Modeling and Forecasting of the Stock Markets

The modeling of financial markets and other social phenomena based on multi-agent systems have attracted the attention of many researchers in recent years (Galla 2006) (Gupta 2001) (Grothmann 2002) (Araújo 2004). A number of phenomena that can be studied with this type of modeling is huge including studies of growth of epidemics, traffic of vehicles or pedestrians and investors in the stock exchange. The modeling of the individuals that make the systems can artificially represent an investor in the stock market, a company in negotiations with suppliers or vehicle on a highway. After modeling the microeconomic aspects of individuals, it is possible to determine the macroeconomic aspect of the whole system. Thus, multiagent systems are able to "capture the complexity and dynamics of behavior and learning in financial markets using models of markets, strategies and more realistic structures" (Farmer 1996 apud Grothmann 2002).

However, these financial models based on multiagent systems have some limitations. There is a semantic specification of the agents (Grothmann 2002) and most models of decision making is done through functions adhoc (Challet 1997; 1999), or mechanisms that can not be adjusted to external data. We have the example of models based on the Minority Game (MG) (Zhang and Challet 1997) who based decisions to buy or sell in a table of binary values that is drawn from random initially. The table has a fixed number of strategies, each strategy has values that represent the last N iterations of the system beyond the act of sale or purchase to be made, so the players choose a strategy based on this table and pay the fixed strategies with scoring positive or negative depending on the outcome of a choice in the learning process by reinforcement. This level of rationality and decision-making does not match with the reality of a real financial market, and not be able to train the staff based on actual market data.

Thus, we have to adapt the decision mechanism of the agents with econometric models as univariate or multivariate linear regression, or models with non-linear regression like neural networks that can be adjusted to real data (Grothmann 2002).

4 Neural Networks

A neural network (NN), also called artificial neural network can be defined as a computational structure designed to mimic in a rudimentary way in which the human brain performs a particular task of interest (Haykin 1999). The main feature of a NN is the ability to learn, through their training. The problem of learning is to find, through an interactive process (where each entry causes a response) and iterative (repeated), a set of free parameters that enable the desired network performance. In other words, it is the adjustment of synaptic weights (*Wi*) and the level of bias that produces the lowest level of error between the desired response and the response estimated by the network (in a time series) or the best boundary of separation (decision) between patterns in a dataset.

In this work we are using neural networks in the decision engine of the agents, to be able to learn from the history of actual market prices and then take a decision on the following period. The main weakness of the NN to forecast future price is the noise in the historical series of data. Therefore, the neural networks are fed with data including stock quotes from the closing price, volumes of trading and moving averages calculated on the closing price of assets.

To implement the neural networks of this work we used the Encog Artificial Intelligence Framework for Java and DotNet (Encog 2009).

5 The Agents

In the method proposed in this work, a population of several agents has been created. For each share of the Bovespa Index we created an agent called from now the decision agent that is dedicated to make the prediction of its own share a step ahead. The decision agent work together with a small population of neural agents that works, in a competitive way, to predict the movement direction of a specific share, always based on its training in actual historical prices of assets. The decision agent decides which neural agent will be used to predict a step ahead. In other words, we have for each share of the index an decision agent connected to a small population of neural agents. Neural agents will use historical market data to train and try to predict one step ahead, and after all the neural agents be trained and tested, the decision agent will capture indication from the most suitable agent.

The last agent system, called from now the central agent has two responsibilities. The first responsibility is to collect the choices that each decision agent has about the direction of each share of the index, after collecting these choices, the central agent will calculate what the final forecast on the index. To generate the final index prediction the central agent will consider two information sets. The first set is composite by each share decision received from decision agents. The second set is the central agent knowledge about the weight of each asset (share) in the index; see Session 2 "The Bovespa Index" from this work for more details about the weights of each asset in the index.

The second responsibility of the central agent is conduct a small population of neural agents that are training based only on historical prices of the Bovespa Index itself. This process is the same that each decision agent makes in every share of the index. Thus, you can perform a comparison of the decision a step ahead of all agents' population from each share and compare with the decision based only on the historical prices of the index. It is expected that the decision-making of the population of agents is more accurate than the decision based only on the price index, since the population of agents looking in each share is more able to capture the microstructure of the market, creating a more realistic estimate.

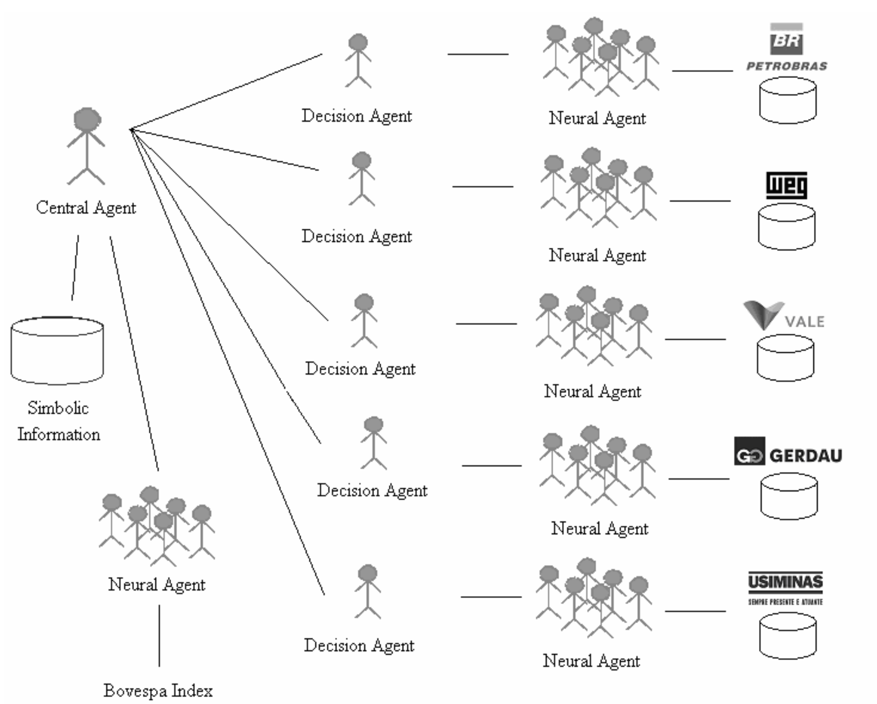


Fig. 2 Population of agents of the system: The decision agents communicate with a small population of neural agents. The neural agents receiving the prices of each asset (share) and the best agent prediction is considered by the decision agent. The central agent, than, communicates with all decision agents to do the final forecast. The central agent considers too its knowledge about the weight of each asset (share) within the index.

6 Prediction a Step Forward

Each *neural agent* will decide the up or down movement a step ahead by a decision engine based in a multilayer perceptron neural network (MLP-NN) acyclic (feedfoward). We used a setup based on 3 layers with resilient backpropagation training algorithm. The last layer always has one neuron that generates a value above or equal 0.5 to indicate high or below 0.5 to indicate low. The second or hidden layer have 5 neurons. Finally, the first layer receives the historical data price and the number of neurons represents the size of the time window used.

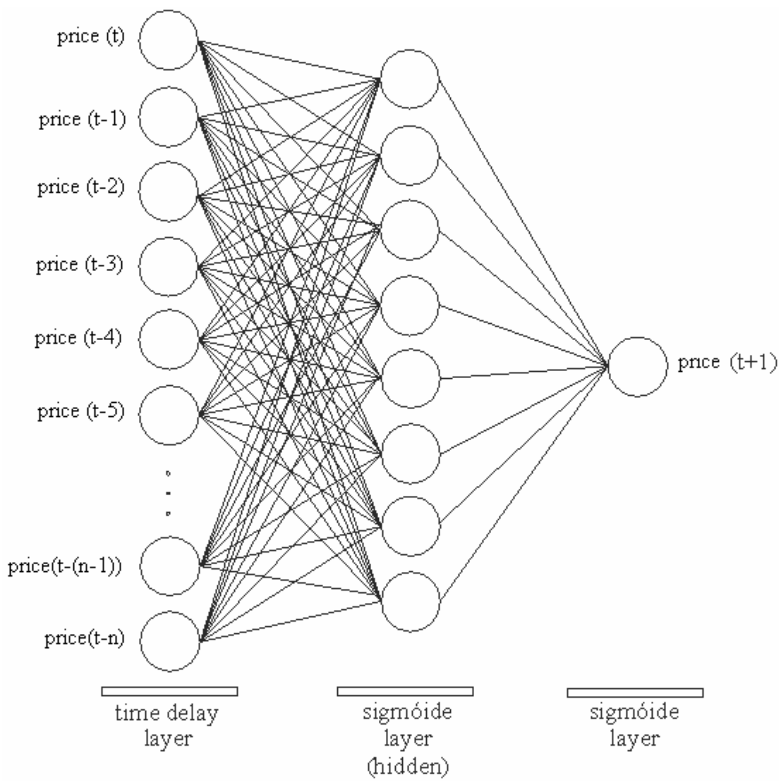


Fig. 3 Configuration of neural network used in the neural agent's engine's decision

It is very important to observe that the final result is a binary prediction, up or down. We used this model to validate our work because another metrics like mean square error didn't represent the best choice for this work. Some results can have a better mean square error but can generate less correct direction indications.

The configuration tested was composed of 10 neurons to 10 days of prices in the first layer. Actually, the neuron receives the daily return, not the price value itself. The return r_t for day t is defined as $\frac{p_t - p_{t-1}}{p_{t-1}}$ where p_t is the actual price of the market on day t.

The intermediate layer (hidden layer) was configured with 5 neurons where the activation function is the sigmoid function, which is defined as a strictly increasing function that displays an appropriate balance between performance linear and nonlinear (Haykin 1999). The last layer formed by one neuron to issue the final decision on the network was configured with a sigmoid activation function too.

All data presented to network was normalized in a range from 0 to 1, the same range of sigmoid function used in the layers. In all tests, we used a data set with three months of daily data. The last ten days were used to test the accuracy prediction of the system and the rest data were used to train de the neural agents´ population. In the test phase just not know data was submitted to the system.

During the training phase, 5000 cycles was done to train the neural agents´ networks, but, to avoid the problem from over-fitting, we used early-stopping. Over-fitting decrease the ability to work with not knows data. Early-stopping training technique is very important to prevent the network from over-fitting. In this work a simple early stopping technique was used. During the training, after each cycle the network power of prediction is tested with training set, and always that the prediction is better or equal than 80%, a cycle counter is started. If, after 100 cycles, the results over training set do not become better, the training stops. The same early-stopping technique was used in all neural agents´ networks, including the neural agents that analyses only the index historical data.

To decide which neural agent will be utilized to predict a stop ahead, the decision agent analysis the training phase from each neural agent, then the central agent takes the decision from the first neural agent that was stopped by early-stopping with 80% or better right results.

Finally, is important to define how de central agent will manage all decisions received from decision agents. The central agent, capture all direction decisions from decision agents and make its own prevision like following:

$$p_{t+1} = \left(\sum_{i=1}^{n} d_i w_i \right)$$

Where, p_{t+1} is the direction prediction (p) in time (t), in other words, p_{t+1} meaning the prediction a step ahead, d_i is the prediction from asset i decision agent, w_i is the weight in the index from asset i, and n is the number from decision agents in the system, actually 66 because we have 66 shares in Bovespa Index.

This formula generates a value that central agent interprets like following: if the result is a value greater or equal than 0.5 then an up direction is expected. If the result is a value smallest than 0.5 then a down movement is expected.

7 Results and Conclusions

Several tests have been done from January, 01/2009 to November, 01/2009. We start the tests in January taking three consecutive months, where the last ten days from last month is used to test the accuracy prediction of the system and the rest data were used to train de the neural agents´ population. After that, we advance one month and go to February, and again, we take three consecutive months to train the system being the last ten days to test the prediction power with not know data. Each period was tested 3 times. Is important to observe that only daily closing prices was used, in others words, only the last price from the day were used to represent the daily price of each asset.

Using the populations from multiagent system, the central agent indicates an average of 60.82% corrects directions. Otherwise, using the same configuration in the neural agents, but using only the data series from the index, the central agent indicates 52.08% of correct directions. The difference of 8.74% in the performance shows that the system proposed by this paper using each share from the index can predict the direction of Bovespa Index better than the same approach based only over the index temporal serie.

Acknowledgements. Thanks to the Statistic and Informatics Department (INE) and the graduate course in Computer Science (PPGCC) of Federal University of Santa Catarina State (UFSC) and the graduate course in Computer Science.

References

1. Araújo, R.M.: Machine learning in complex multi-agent systems: a case study in an environment under bounded rationality. PPGC UFRGS, Porto Alegre (2004)
2. Bar-yan, Y.: Dinamics of Complex Systems. Addison-Wesley, Massachusetts (1997)
3. Bovespa. BMF&Bovespa. A bolsa do Brasil (2009), `http://www.bovespa.com.br` (accessed February 1, 2009)
4. Challet, D., Zhang, Y.: Emergence of Cooperation and Organization in an Evolutionary Game. Institut de Physique Théorique, Université de Fribourg, 1700 Fribourg, Switzerland (1997)
5. Challet, D., Marsili, M., Zhang, Y.: Modeling Market Mechanism with Minority Game. Institut de Physique Théorique, Université de Fribourg, CH-1700 (1999)
6. Encog. Encog Artificial Intelligence Framework for Java and DotNet (2009), `http://www.heatonresearch.com/encog` (accessed November 10, 2009)
7. Fama, E.F.: Efficient Capital Markets: A Review of Theory and Empirical Work. Journal of Finance 25(2), 383–417 (1970)
8. Fama, E.F.: Efficient Capital Markets II. Journal of Finance 46(5), 1575–1617 (1991)
9. Farmer, J.D., Lo, A.: Frontiers of Finance: Evolution and efficient markets. Proceedings of the National Academy of Sciences 96, 9991–9992 (1996)
10. Galla, T., Mosetti, G., Zhang, Y.-C.: Anomalous fluctuations in Minority Games and related multi-agent models of financial markets. The Abdus Salam International Center for Theoretical Physics, Strada Costiera 11, 34014 Trieste, Italy (2006)

11. Grothmann, R.: Multi-agent market modelling based on neural networks. Faculty of Economics, University of Bremen (2002)
12. Gupta, N., Johnson, N.F., Hauser, R.: Robust Methods for Tracking Intelligent Agents Playing in an Artificial Financial Market. In: The Sixth Intl. Joint Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS 2007), 1108a (2007)
13. Haykin, S.: Neural Networks: A Comprehensive Foundation, 2nd edn. Prentice Hall, New Jersey (1999)
14. Malkiel, B.G.: A random walk down Wall Street, 7th edn. W. W. Norton & Company, New York (1999)
15. Zhang, Y., Challet, D.: Emergence of Cooperation and Organization in an Evolutionary Game. Physica A 246, 407 (1997)

Experiments on Robotic Multi-agent System for Hose Deployment and Transportation

Ivan Villaverde, Zelmar Echegoyen, Ramón Moreno, and Manuel Graña

Abstract. This paper reports an experimental proof-of-concept of a new paradigm in the general field of Multi-Agent Systems, a Linked Multi-component Robotic System. The prototype system realizes a basic task in the general framework of a multi-robot hose transportation system: the trasportation along a linear trajectory. Even this simple task illustrates some complexities inherent to the general task of hose transportation. Artificial Vision is used to perceive the state of the system composed of the agents and the hose. The robotic agents are autonomously controlled by means of an scalable control heuristic. The system is able to deploy and transport a passive object simulating a hose in straight line, avoiding the formation of loops and dragging between robots.

1 Introduction

Multi-Agent Systems (MAS) have been proposed in several application domains as a way to fulfill more efficiently a task by cooperation between several autonomous agents [7]. This paradigm has a very direct application in robotics, as the physical limitations of the real-life robots and the environments they are supposed to work in impose severe restrictions to their capability to fulfill some tasks, up to the extent that there are complex tasks that can not be accomplished by a single robot and must be performed necessarily by a multi-robot system [2].

In the last two decades a lot of effort has been put in transferring the MAS paradigm to mobile robotics. There are several reviews giving different categorizations [2, 1, 6, 3] focusing on different aspects of the multi-robot systems. Recently, in [3] a categorization of Multi-component Robotic Systems (MCRS) has been done focusing, among other aspects, on the way the robotic agents are physically connected, identifiying three main types of MCRS: Distributed, Linked and Modular.

Ivan Villaverde · Zelmar Echegoyen · Ramón Moreno · Manuel Graña
Computational Intelligence Group, University of the Basque Country (UPV/EHU)
e-mail: www.ehu.es/ccwintco

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 573–580.
springerlink.com

This categorization presents an interesting novelty: while the Distributed and Modular MCRS are familiar concepts, representing groups of robots unlinked and joined by a rigid component, respectively, the Linked MCRS is a new category, not previously identified in the literature and characterised by a linking passive element between robots. This new category theoretically shows some new issues coming from this passive element that the system's agents have to cope with and we are starting to deal with them from several points of view. In [5, 4] we addressed the problem of modelling and derived the formal inverse kinematics and dynamics of this kind of robots. In this paper we dwell more on the physical realization of a proof-of-concept of a Linked MCRS with a concrete set of robots and a piece of electric cable as the passive linking element.

In the following section 2 a brief description of the hose transportation problem is given. In section 3 we detail the specifics of the approach followed to implement this proof-of-concept, giving a description of the artificial vision perception system and the agents' control heuristic, finishing with an example of operation. Finally, in section 4 we discuss the specific traits of this kind of multi-agent robotic systems identified through the implementation of this proof-of-concept.

2 Robotic Multi-agent System for Hose Transportation

The transportation, deployment and manipulation of a long[1] almost uni-dimensional object is a nice example of a task that can not be performed by a single robotic agent. It needs the cooperative works of a team of robots. In some largely unstructured environments like shipyards or large civil engineering constructions a typical required task is the transportation of fluid materials through pipes or hoses. The manipulation of these objects is a paradigmatic example of a Linked MCRS, where the carrier robot team will have to adapt to changes in the dynamic environment, avoiding mobile obstacles and adapting its shape to the changing path until it reaches its destination.

The general structure and composition of this hose transportation robotic MAS would be that of a group of robots attached to the hose at fixed or varying points. The robots would search for space positions in order to force the hose to adopt a certain shape that adapts to the environment, while trying to lead the head of the hose to a goal destination where the corresponding fluid will be used for some operation. The changing environment conditions may force changes in the hose spatial disposition, to which the robots should be able to react. This general form can take multiple implementations depending on several elements of its design:

- Robot-hose attachment: Robots could be fixed to a point of the hose, they can move along it, or they can pull it through special gripping mechanisms.
- There can be a centralised control for all of the robots or, in a true MAS approach, it can be decentralised, each robot taking its own control decision.

[1] The adjective "Long" used here is relative to the size of the individual robots. The object's length must be some order of magnitude greater than the robot's size.

- Robots can be homogeneous or heterogeneous, having different configurations and tasks (e.g., "pulling" robots, which tow the hose, and "cornering" robots, which take fixed positions and give shape to the hose).
- Perception can be global, with some agent acquiring a global view of the system, or local, with every robot acquiring information of its close surroundings.

Our research group is actively involved in studying the hose manipulation problem from diverse points of view, producing modelling and simulation of the hose-robots dynamics and some problem characterizations [5, 8, 4]. However, the starting point to achieve this system on a real robotic platform is the physical realization using real robots of a vision controlled robotic MAS which faces the basic non-trivial issues in this hose transportation problem.

3 Proof-of-Concept Prototype and Experiment

For the physical realization of this proof-of-concept we have defined the basic problem to solve: to control a robotic MAS whose objective is to perform the transportation of the hose in a straight line in an environment without obstacles from an initial arbitrary configuration of hose and robots. In case that the robots are fixed to the hose or that an individual robot is not powerful enough to pull it or the initial configuration of the hose is arbitrary, this task has to be necessarily performed by several robots. Several robots have to be controlled to guarantee a certain hose configuration and each individual robot motion has to be controlled in order to keep a desired formation. Thus, although the task is the simplest one that can be defined, it is a non-trivial task which poses several problems whose solutions are the cornerstones of the solution of more sophisticated ones. Besides, we will need to deal with the restrictions that the real robot's embodiment impose, which must be coped with in order to obtain a working system realizing the proposed task.

The task defined above has rather diverse solutions depending on the actual robots employed and the actual physical features of the passive element. In this section we give an account of the hardware used, the image analysis procedures applied to obtain the visual feedback and the control heuristic applied to define the control strategies.

3.1 Hardware and Communications

The experimental solution to this problem was implemented using three small SR1 educational robots. Each robot was attached to an electrical cable of 1 cm. of diameter, which takes the place of the hose, by means of a bearing which allows the robot to rotate freely under it. One camera was placed on a 2.5 meters high mobile stand, facing down at an angle of around 60^o and capturing about 2-3 meters of the floor in front of it. The camera was attached to one laptop PC which performed the centralised perception and control processing. Control commands were sent to the robots using a relatively noisy RF wireless channel. The robot's compass information is used in the "follow the leader" strategy described below.

3.2 Perception

The centralised perception is provided by a single camera that captures the scene encompassing the three robots and the hose. The images acquired are segmented in search for the three robots and the hose. This segmentation process assumes several conditions on the environment's configuration: blue robots, dark (non-blue) hose, uniform floor of bright color (non-blue) and white, uniform illumination. Image segmentation is composed of two separated processes, one for the detection of the robots and the other for the localization of the hose.

Robot's Segmentation

For the segmentation of the robots we are mainly interested in avoiding the effect of strong reflections on the floor and enhance the image color contrast in order to bring out the blue robots from the bright floor. This is achieved by means of a preprocessing step in which a Specular Free (SF) image [10] is created. We follow the Dichromatic Reflection Model (DRM) [9], where images are the sum of two components: the diffuse component (which models the chromacity of the observed surfaces) and the specular component (which models the chromacity of the light source which illuminates the scene). Assuming a white light source, in this algorithm we profit from the characteristics of the RGB cube, as pixels corresponding to reflections (and bright surfaces close to white color) will be very close to the gray color axis that goes from point (0,0,0) to point (1,1,1) of the RGB cube while pixels corresponding to diffuse components in the image will move away from it and will be closer to the pure color axes. This property is used to reduce the intensity of the specular pixels and improve the intensity of the diffuse ones proportionally to their distance with the grayscale axis. Given an input RGB image $X = \{x(i,j)\}$, where $x(i,j) = \{r_{ij}, g_{ij}, b_{ij}\}$, a chromatic image $C = \{c(i,j)\}$ is computed as

$$c(i,j) = \max(r_{ij}, g_{ij}, b_{ij}) - \min(r_{ij}, g_{ij}, b_{ij}), \quad (1)$$

in this equation, using normalised values, white/gray pixels will have value $c(i,j)$ close to zero, while colored regions, corresponding to diffuse components, will be close to one.

The RGB image is then transformed to HSV space and its intensity channel is replaced with the computed chromatic image C, so that white/gray pixels become very close to black. This HSV image is then transformed back to RGB space. Since we are looking for blue robots, they can be easily found in the SF image looking for the regions with highest intensities in the B channel. The result of this step is a collection of boxes $\mathbf{R} = \{R_1, \ldots, R_n\}$ giving the regions of the image containing the location of the robots. Robot position p_i will be the centroid of that region. When processing a sequence of images, the robot detection process is done in the neighborhood of the previous image detected boxes.

Hose Segmentation

The segmentation of the hose takes advantage from the strong contrast of a dark object over a bright floor. Given the original RGB frame and the regions obtained from the robot's segmentation, hose's segmentation is performed by the following processing steps: (a) The image is binarized, white pixels code the hose detection. (b) The binary image is skeletonized. (c) Regions of the skeletonized binary image are identified and labeled. (d) Discard very small regions. (e) Discard regions that do not connect two of the boxes found before containing a robot. Each region obtained after this process is considered a segment of the hose, we denote them $\mathbf{S} = \{S_1, ..., S_{n-1}\}$ where segment S_i connects robot boxes R_i and R_{i+1}.

3.3 Control Heuristic

Due the limited computing capabilities of the robotic platform used, the control commands are determined in a separate single computer and then communicated to the robots. However, each of the actions of the robots is computed independently, without taking into account the state of the other robots, as if they were computed by each of the independent agents. Each robot's control will be determined only by the perception of the segment of the hose that is immediately ahead of it and the information about the orientation of the leader. In this way the system is very scalable and can be extended to any number of robots.

The trajectory of the robots will be controlled by a "follow the leader" strategy. The leader will be remotely controlled and the remaining robots will follow its orientation: at each time step, the tema robots will check if they have the same orientation as the leader, using their compasses. They will reorient themselves trying to align with the leader in case they are not.

Each robot's speed will be given by a control heuristic that takes into account the state of the hose segment ahead of it. This state is a function of the curvature of the hose segment. Given an image hose segment $S = \{s_1, ..., s_m\}$, where s_j is a pixel site coordinates, we define the curvature c of the segment S as the proportion between the maximum distance d_h from the hose segment points s_j to the line L_{p_1,p_2} that crosses both robot's positions (p_1, p_2) and the distance between the robots d_r:

$$c = \frac{d_h}{d_r}, \tag{2}$$

where $d_h = \max_i \left\| s_i - L_{p_1,p_2} \right\|$ and $d_r = \|p_1 - p_2\|$.

This is equivalent to obtain the ratio of the sides of the rectangle that encloses the hose segment and has the length of the distance between the robots. Being it a ratio value, it is not so influenced by the perspective.

This heuristic underlying reasoning is that if two robots are too close, the hose segment between them will fold and increase its curvature, with the risk of forming loops. The robot at the rear of the hose segment must reduce its speed. On the other hand, if the two robots attached to the hose segment are separated enough the hose segment will be very close to the straight line. A segment too tight will produce dragging between the robots. In this case, the rear robot should accelerate to ease the tension of the hose. Three rules determining the consequent robot speed were defined over the values of this proportion c:

- $c \leq 0.15$: The hose segment is too tight. The rear robot takes *fast* speed trying to shrink the hose to avoid dragging the front robot.

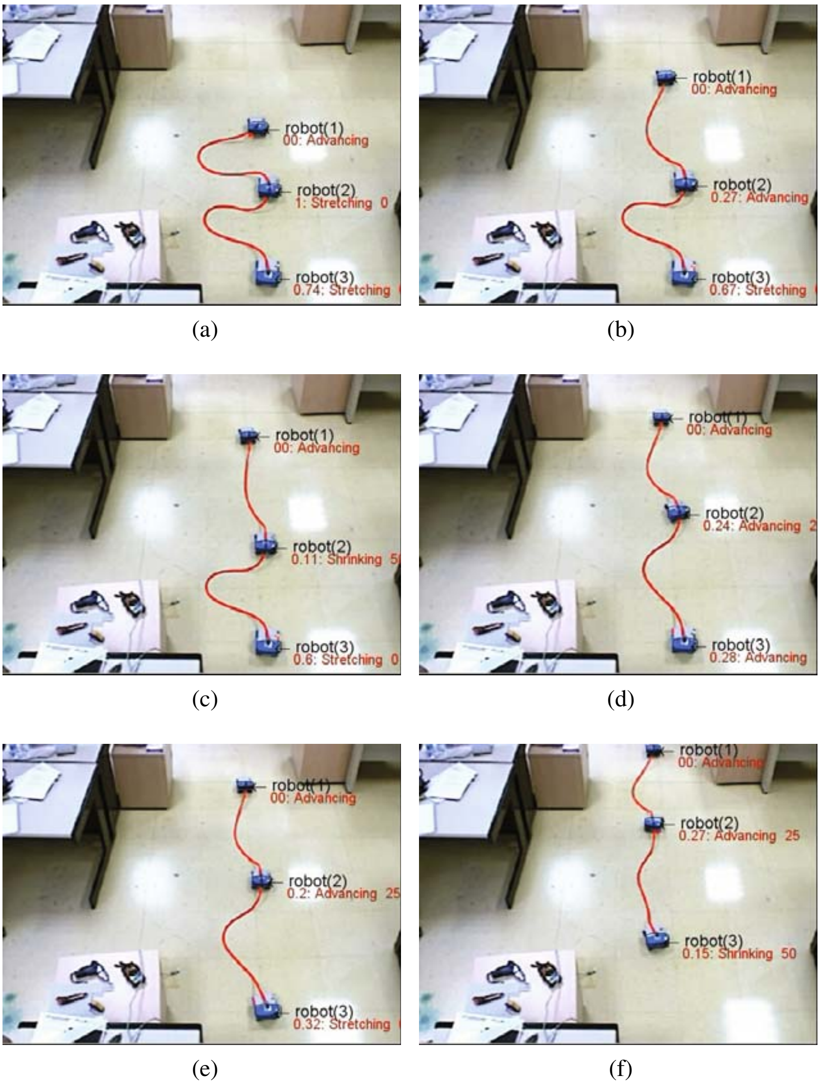


Fig. 1 Frames extracted from the video of an example realization of the hose transportation task

- $c \geq 0.30$: The hose segment has shrunk too much. The rear robot *stops* and waits for the hose to stretch to prevent the formation of loops.
- $c \in (0.15, 0.30)$: The hose has the correct length. The rear robot takes *cruise* speed and continues advancing.

3.4 Experiment Realization

In figure 1 an example of the realization of the hose transportation task defined above is shown. In each frame, the robots are marked with the action they are taking and the curvature of their respective hose segment. Detected hose segments were marked in red in the original colored video. The six frames are extracted from the video generated in the test and show how the system reacts to the different states that the hose takes:

- Figure 1a: Starting position. The leader starts towing the hose, while the 2nd and 3th robots wait for it to stretch enough.
- Figure 1b: The first segment's curvature falls below 0.30 ($c_1 = 0.27$). The 2nd robot starts advancing at cruise speed. The 3th robot keeps waiting ($c_2 = 0.67$).
- Figure 1c: The first segment is too stretched ($c_1 = 0.11$). The 2nd robot accelerates to fast speed to shrink it. The 3th robot keeps waiting ($c_2 = 0.6$).
- Figure 1d: The first segment's curvature is within limits ($c_1 = 0.24$), the 2nd robot brakes itself to attain cruise speed. Second segment's curvature also enters within limits ($c_2 = 0.28$), therefore the 3th robot starts advancing at cruise speed.
- Figure 1e: Second segment raises again above 0.30 ($c_2 = 0.32$), the 3th robot stops. The 2nd robot keeps advancing at cruise speed ($c_1 = 0.2$).
- Figure 1f: Second segment falls below limits ($c_2 = 0.15$), the 3th robot accelerates to fast speed. The 2nd robot keeps advancing at cruise speed ($c_1 = 0.27$).

Some videos can be found on our web site: http://www.ehu.es/ccwintco/index.php/DPI2006-15346-C03-03-Resultados-videos-control-centralizado.

4 Conclusions and Discussion

We have realized the physical proof-of-concept of the vision based control of a robotic MAS, in the form of a Linked MCRS, performing the transportation of a hose-like object. The experiment of physical realization of the hose transportation task has served also to demonstrate some specific traits of the hose-robot Linked MCRS. For such a simple behavior as the "follow the leader", we have observed that the hose-like passive element introduces dynamic interaction problems, as the hose may be an obstacle for the robots, can drag them or provoke dragging between them and its weigh and rigidity impose motion constraints to the robots to follow a desired path.

The hose-like element introduces new perception and measurement needs: we need to observe (segment) the hose, and to compute some measure of its state that allows to build a system of rules that determines the agents' behavior as a function of this state. The hose imposes also an ordering on the motion of the robots, both spatial and temporal. The perception of the hose and the fine tuning of the effects of its state are problems not shared by other MCRS paradigms. The experiment realization has been a success in the sense of demonstrating the inherent features of Linked MCRS and their needs.

The long term objective is the realization of a fully distributed control for a robotic MAS performing the deployment and transportation of a hose in a dynamic environment. However we need to define a collection of tasks that gradually approaches this goal. We are currently working on this and on the physical realization of the robot-hose system with other robotic modules.

References

1. Uny Cao, Y., Fukunaga, A.S., Kahng, A.: Cooperative mobile robotics: Antecedents and directions. Autonomous Robots 4(1), 7–27 (1997)
2. Dudek, G., Jenkin, M.R.M., Milios, E., Wilkes, D.: A taxonomy for multi-agent robotics. Autonomous Robots 3(4), 375–397 (1996)
3. Duro, R.J., Graña, M., de Lope, J.: On the potential contributions of hybrid intelligent approaches to multicomponent robotic system development. Information Sciences (accepted for publication)
4. Echegoyen, Z.: Contributions to Visual Servoing for Legged and Linked Multicomponent Robots. PhD thesis, University of the Basque Country (2009)
5. Echegoyen, Z., d'Anjou, A., Villaverde, I., Graña, M.: Towards the adaptive control of a multirobot system for an elastic hose. In: Köppen, M., Kasabov, N., Coghill, G. (eds.) ICONIP 2008. LNCS, vol. 5506, pp. 1045–1052. Springer, Heidelberg (2009)
6. Farinelli, A., Iocchi, L., Nardi, D.: Multirobot systems: a classification focused on coordination. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 34(5), 2015–2028 (2004)
7. Ferber, J.: Multi-agent systems: an introduction to distributed artificial intelligence. Addison-Wesley, Reading (1999)
8. Lopez-Guede, J.M., Graña, M., Zulueta, E., Barambones, O.: Economical implementation of control loops for multi-robot systems. In: Köppen, M., Kasabov, N., Coghill, G. (eds.) ICONIP 2008. LNCS, vol. 5506, pp. 1053–1059. Springer, Heidelberg (2009)
9. Shafer, S.A.: Using color to separate reflection components. Color Research and Aplications 10, 43–51 (1984)
10. Tan, R.T., Ikeuchi, K.: Separating reflection components of textured surfaces using a single image. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(2), 178–193 (2005)

On Cooperative Swarm Foraging for Simple, Non Explicitly Connected, Agents*

Mireia Sempere, Fidel Aznar, Mar Pujol, and Ramon Rizo

Abstract. Nowadays there are several applications that use swarm robotics for solving research tasks and resource exploitation. Most of these applications are based on complex agents that require explicit communication between them. These systems are difficult to introduce in certain environments because of these features, where agents can not always communicate between them and where it would be necessary a large swarm. This paper presents a swarm system for a collective resource exploitation. The main features of the agents of this system are their simplicity and they do not communicate with each other in explicit way. A microscopic model that shows the individual performance of agents has been proposed, and a macroscopic model that describes the overall swarm system has been provided. Several tests that show the convergence of the swarm towards the best resource in an unknown environment have been analyzed.

1 Introduction

Swarm robotics is a decentralized approach to coordinating multi-robot systems, where the collective behaviour emerges from local interaction between agents and their environment. It is based on behaviours observed in social insects (ants, termites, bees, wasps) which are great examples of how a large number of individuals can interact to create intelligent collective systems. This type of decentralized coordination allows the system to develop its global objectives embedding simple behaviour patterns, possibly inspired by nature, in each individual agent that form the

Mireia Sempere · Fidel Aznar · Mar Pujol · Ramon Rizo
Department of Computer Science and Artificial Intelligence
University of Alicante
e-mail: {mireia,fidel,mar,rizo}@dccia.ua.es

* This work has been supported by Conselleria d'Educacio de la Generalitat Valenciana, project GVPRE/2008/040, University of Alicante, project GRE08P02 and Ministerio de Ciencia e Innovacion, project TIN2009-10581.

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 581–589.
springerlink.com

system. Its main advantage is robustness, which manifests itself in different ways. First, swarms are a relatively simple set of agents, which do not have pre-assigned any task or specific role. Therefore, the swarm can reorganize itself dynamically or reorganize the way an agent is used in the system. Secondly, and for the same reasons, a swarm is high tolerant to the failure of an individual agent. Third, the fact that control is completely decentralized makes that there is no point of failure or vulnerability in the swarm. Although swarm robotics is a relatively recent research field, an extensive collection of essays can be found [1, 2, 3, 4, 5, 6, 7, 8]. The potential of this approach is considerable: any task in which there are a number of physically distributed objects that need to be explored, inspected, collected, completed, rescued or mounted on structures is a potential application of swarm robotics.

The collective exploitation of resources (foraging) is an application particularly relevant in this type of system. This application requires on the one hand, the exploration of the environment to look for a specific type of resource, and on the other hand, to extract those resources and use or transport them to the base station of the swarm. There are several approaches that use swarm robotics to solve this task. Most of these are based on agents that are able to communicate with other agents of the swarm [3, 9] (or by direct communication with the rest of the swarm or using indirect methods such as artificial pheromones). In such applications a homogeneous group of robots is usually used, normally small, with complex sensing systems (such as laser or vision) [10, 7, 1, 11].

The main purpose of this paper is to provide a swarm system without explicit connection (where the agents of the system can not communicate with each other) based on simple behaviours, which is able to locate dynamically the most promising resource in an unknown environment. The aim of the agents that form the system is to identify the resources (other type of agents are responsible to collect them). The paper is focus on developing a swarm oriented to resource location and not to transport or collect them.

2 Swarm Design

The game proposed in (`http://icosystem.com/game.htm`) presents an example of the use of extremely simple agents. In this game an agent can develop three types of behaviour (refugee or fugitive, defender or aggressor). These behaviours are the base of complex behaviours of all the group. These behaviours are discussed in [4], which shows that from these simple behaviours can be obtained three different behaviours of the swarm: expansion, cycle or aggregation (at some indeterminate point in the environment[1]. Obviously, with this kind of behaviours is not possible to perform tasks of exploration and exploitation of resources.

[1] Although the rules that define the operation of the agents are extremely simple, it is not the case of the operation of all the swarm. As discussed in [12], design and evaluation of the behaviours that generate self-organization in swarm systems is a complex problem that depends on many factors, including the initial position of agents, the distribution of the environment.

Drawing on these studies, the extension of some behaviours explained in [4] is proposed in this paper. Two specific behaviours will be used, aggressive agents, that pursue other agents, and elusive agents that flee from their aggressors. As discussed in [4], if each agent pursues a different agent, then the swarm tends to converge at a point in space. By contrast, with the behaviour of elusive agents, the swarm tends to expand infinitely.

An agent $r_i \in R$ is defined as a vector $r_i = (s_i, p_i, a_i)$ which contains states $s_i \in S$, sensors $p_i \in P$ and actuators $a_i \in A$. Each agent uses two actuators $a_i = (v_{tras}, v_{rot})$ that allow the agent to establish its translational and rotational speed. And each agent has 3 sensors $p_i = (m_i, c_i.t_i)$, where:

- m_i: Sensor that allows an agent to identify its partner in the swarm. Each robot has a partner, such that $\forall m_i \in M, \forall m_j \in M, i \neq j, m_i \neq m_j$.
- $c_i \in [0,1]$: Sensor that indicates the quantity of resource that exists in the current position of the robot.
- $t_i \in [0,\infty[$: Internal clock of the robot, it allows to know the current time.

An agent may have two states $s_i \in \{expand, group\}$: *expand* corresponds to the behaviour of elusive agents, where the agent r_i has to calculate the repulsion vector respect to the robot m_i; and *group* corresponds to attraction behaviour, where the agent has to calculate the attraction vector to m_j[2]. The change from one state to another is probabilistic and is governed by three factors listed below:

Energy of the agent e. The initial energy of each agent is e_{init} that decreases with time. The energy is defined such that $e = e_{init}\left(-\frac{-e_{init}+\Delta_e}{e_{init}}\right)^t$, being Δ_e the increase in energy per time unit t.

Probability of cluster p_a. The probability that a particular agent changes to state *group*. As the energy of the agent decreases, the probability of cluster must be higher since we are interested in the swarm tends to converge in the position of the environment where there is more amount of resources. Thus p_a is defined as: $p_a = 1 - \frac{e}{2e_{init}}$.

Probability of expansion p_e. The probability that a particular agent is in the state *expand*. It is defined as the inverse of the probability of cluster ($p_e = 1 - p_a$).

The behaviour of individual agents will be manage by a probabilistic binary state machine (with two states: *expand*, *group*), with transition probabilities p_a and p_e, with initial state *expand*, from which each agent will calculate its state at time t. An agent can only change its state from time to time t_c, to allow the execution of behaviours. This time is constant and identical among agents.

In addition, when an agent is on top of a resource, it is interesting that its speed is slower to inspect the resource and to indicate its position to the rest of the swarm.

[2] The simplest way to calculate this vector requires knowing the current position of the robot pos_i and the position of its partner pos_{mi}, so that the repulsion vector can be defined as $v_e = pos_i - pos_{m_i}, \hat{v}_{expand} = \frac{\mathbf{v}_e}{\|\mathbf{v}_e\|}$ and the attraction vector as $\mathbf{v}_g = pos_{m_i} - pos_i, \hat{v}_{group} = \frac{\mathbf{v}_g}{\|\mathbf{v}_g\|}$. Nevertheless, it is possible to get the attraction or repulsion vector without knowing the current position of robot, using only indirect methods of location.

Thus, we propose to obtain a new vector $\mathbf{v} = \mathbf{v}_b \times c^k$, where k is a modifier of the intensity of attraction of the resource and $\mathbf{v}_b$ is the movement vector obtained from the behaviour s_i of the robot. Starting at $\mathbf{v}$ we obtain the rotational and translational speed of the robot, (v_{tras}, v_{rot}).

Moreover, if an agent collides with another agent, it will generate a random velocity vector $\mathbf{v}_{rand}$ until the end of the collision. This system is simple but it helps to avoid situations in which a calculation of collision avoidance can require a completed knowledge of the nearby environment of the agent, which is not possible given the limited perceptual characteristics of these agents.

Finally, the initial position of the swarm is located forming a matrix of a maximum of 10 columns by an undetermined number of rows (depending on the size of the swarm) such that $pos_{init}(r_i) = \{i \pmod{10}, \lfloor \frac{i}{10} \rfloor\}$, so the partner of each agent is defined as $m_i = r_a, a = i + 1 \pmod{|R|}$.

2.1 Macroscopic Model

Once the microscopic model of each agent has been defined, we want to determine which behaviour will have the swarm. Specifically this paper focuses on analyzing the convergence of the swarm in a homogeneous group. Initially, the basic behaviour, the swarm will tend to, will be determined.

The behaviour of *group* and *expand* are controlled by the probability of cluster p_a and expansion p_e respectively. Therefore, to determine whether the agents of the swarm tend to a certain state, we are interested in the number of agents in each cycle $\Phi = \left\lfloor \frac{t}{t_c} \right\rfloor$ that develop each of the above behaviours. Assuming that initially both behaviours have the same probability:

$$\begin{aligned}
&N_e(0) = \tfrac{1}{2}|R| \\
&N_e(\Phi) = p_e \cdot N_a(\Phi - 1) - p_a \cdot N_e(\Phi - 1) \\
&N_a(0) = \tfrac{1}{2}|R| \\
&N_a(\Phi) = p_a \cdot N_e(\Phi - 1) - p_e \cdot N_a(\Phi - 1)
\end{aligned}$$

$|R|$ is the number of robots of the swarm, N_a the number of robots that are in the state *group* in the cycle Φ and N_e is the number of agents in the state *expand* in the cycle Φ. Taking into account the value of the probabilities p_a and p_e, a generative function have been calculated based on the previous recurrence equations[3] which allows us to calculate the number of agents in a particular state:

$$N_e(\Phi) = \frac{1}{2} \frac{|R|(1 + \left(-\frac{-e_{init} + \Delta_e}{e_{init}}\right)^{\Phi} \Phi)}{\Phi + 1}; N_a(\Phi) = |R| - N_e(\Phi) \tag{1}$$

[3] A more general method to calculate the number of robots that are in a state is to obtain the closed form of the previous recurrence equations. However, in this case, the generative equation proposed from these equations produces smoother transitions than those obtained by the closed form of the recurrence equations.

Using the above equation we can determine when all the agents tend to develop an aggregation behaviour and therefore, as shown in [4], when the swarm will be grouped at a certain point (this occurs when $N_e(\Phi)$ is close to 0). Thus, for the given swarm model, we can calculate when the system will converge in a given area, knowing the number of agents, the initial energy of the agents and their decrement per cycle. It is easy to check that:

$$e_{init} > \Delta_e, \Delta_e > 0, \lim_{\Phi \to \infty} \frac{1}{2} \frac{|R|\left(1 + \left(-\frac{-e_{init}+\Delta_e}{e_{init}}\right)^{\Phi} \Phi\right)}{\Phi + 1} = 0 \qquad (2)$$

3 Experimentation

MASON [13] has been used for the simulation of the system. It is a simulator of multi-agent systems, powerful and versatile. Figure 1 shows a screenshot of this tool.

A continuous 2D environment has been developed based on MASON. This environment can contain discrete cells of resources, where these cells have the same size as an agent (1x1m). The amount of resources in a cell is represented progressively from white (no resource) to black (maximum amount of resources). For each simulation we can delimit the maximum size of the environment (for each test must be specified if the environment is delimited or not).

Five maps of resources have been proposed to carry out the tests. The first map do not have any resource and it is not delimited in size. The other four maps are shown in Figure 1b.

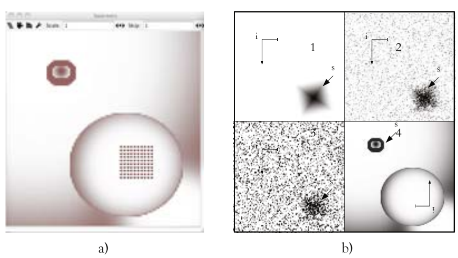


Fig. 1 Screenshot of MASON simulator. Robots are represented by circles of 1x1 meters, where the direction is indicated by a line b) Maps used for tests, *i* indicates the initial position of robots, *s* indicates the major resource in the environment.

As shown in the figure, a simple map $b1$ has been defined, where resources are located in an area of 30 x 30 meters and there is only one resource area. $b2$ is the same map with a 25% of Gaussian noise in an area of 100 x 100 meters, at $b3$ with 50% of noise. $b4$ is a more complex map, with different resource areas, of 100 x 100 meters.

3.1 Design of Experiments

This section focuses on testing the macroscopic model presented, and in assessing the behaviour of obtaining resources depending on the size $|R|$ of the swarm and on the noise of the environment. Various tests described below have been developed. Unless otherwise specified all tests use the following parameters: $(|R|, t_c, e_{init}, \Delta_e) = (100, 200, 1000, 10)$.

The aim of the first test is to check the macroscopic model proposed in section 2.1. Five simulations have been performed for a map without resources, using infinite-sized environments. The purpose of this test is to determine the coincidence of macroscopic model with the simulation of the swarm, and to check the number of iterations needed for the convergence of the system in an environment without resources.

The second test is designed to determine the trend of the swarm to find the most promising resource, assessing the importance of the number of robots used. 5 simulations have been performed for each swarm size $|R| = \{10, 25, 50, 100\}$ on map $b4$. For each simulation, the point marked by the agents (centroid of the swarm) will be determined as a resource centre. The environment will be limited to a size of 100x100m.

The third test checks the robustness to noise of the swarm system. 5 simulations of aggregation for each of the maps have been performed. Specifically, these simulations were carried out using the map 1 (without noise), 2 (25% of Gaussian noise) and 3 (50% of Gaussian noise). Maps with finite size were used (100x100m).

3.2 Results

Figure 2 shows the results of the tests to check the macroscopic model. As can be seen the macroscopic model and the simulation tend to generate more agents in the state *group* as increasing t. Similarly, the deviation of the swarm[4] tends to 0 with the number of iterations.

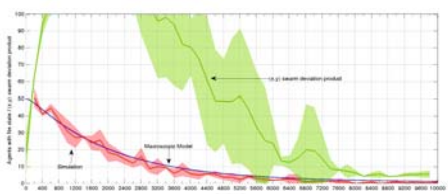


Fig. 2 Results of experiments on a map without resources to verify the macroscopic model. It shows the average (dark line) and variance (clear zone) of each curve.

[4] The deviation of the swarm is calculated as the product, for each of the axes (x, y), of the standard deviation of the position of agents.

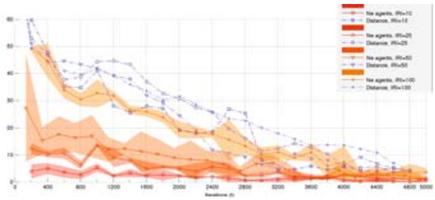


Fig. 3 Euclidean distance of the swarm to the source, together with the number of agents in state *expand* for each of the following sizes of swarm $|R| = \{10,25,50,100\}$. The average of the Euclidean distance for each swarm size is represented. For the number of agents in state *expand*, the average (dark line) and variance (clear zone) of each curve are shown.

Figure 3 shows the results of 5 simulations for different swarm sizes $|R| = \{10,25,50,100\}$ in map $b4$. It can be seen how the distance to the optimal resource is reduced with number of iterations t to arrive practically to 0, which means that the swarm is located on the resource.

Figure 4 presents the results of robustness of the swarm with regard to noise. It shows the optimal distance of the resource and the deviation of the swarm.

4 Discussion

As can be seen in figure 2 the model presented in equation 1 correctly predicts the number of agents in a certain state, coinciding with the results obtained using the simulator. As discussed above, an aggregation behaviour of agents that develop a *group* behaviour stabilizes within a limited time as long as the connection between agents form a connected graph as is shown in [4]. Therefore, as the proposed swarm tends to make that all agents have an aggregation behaviour (see equation 2) and is connected, we can say that the swarm will tend to cluster when the number of agents in the state $N_e(\Phi) \approx 0$.

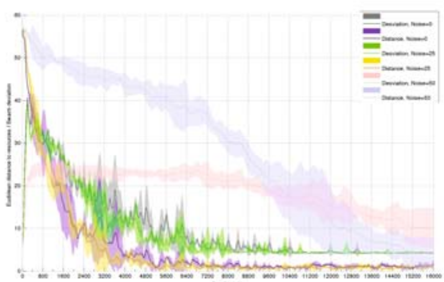


Fig. 4 Deviation of the swarm and the optimal distance to the resource for different levels of Gaussian noise

Furthermore, figure 3 shows as the distance from the centre of mass of the swarm to the most important resource area decreases until be practically 0, regardless of the number of agents used. Similarly the deviation of the swarm also decreases progressively, grouping the agents around those resources. Several tests have been carried out (20 using the map $b1,2,3$ for swarms of 100 robots and 20 for swarms of different sizes using the map $b4$, which has several local maxima that make the search difficult for the swarm). In all these tests, a clear trend towards resource is obtained. Analytically determine the point at which will converge the swarm is a complex task that will not be presented in this paper. However we consider that the information provided demonstrates empirically the aggregation of the swarm in one of the most promising resource locations. Although in all tests carried out, the optimal resource has located, in this task of search and exploitation is not essential to locate the best resource (often it is enough to find a large resource).

The robustsness of the swarm to noise has been proved. For that, the map $b1$ has been modified adding a 25% (map $b2$) and 50 % (map $b3$) of Gaussian noise. Figure 4 shows the results of the original map and the maps with noise. For the map with a 25% of noise the convergence of the swarm is almost the same as for the map without noise. For the map with a 50% of noise the convergence towards the area of more resources is slower but constant.

This paper has provided an extremely simple swarm of agents without explicit connection. A macroscopic model that verifies the convergence of the swarm has been presented and it has been tested experimentally in a satisfactory way. In addition, several tests that verify the convergence of the swarm even in environments with noise have been carried out. Therefore, we consider this study as a first approach to the design of a robust swarm to perform tasks of exploration and collection of resources in unknown environments.

References

1. Hayes, A.T., Martinoli, A., Goodman, R.M.: IEEE Sensors Journal 2(3), 260 (2002), doi:10.1109/JSEN.2002.800682; Special Issue on Artificial Olfaction, Nagle, H.T., Gardner, J.W., Persaud, K. (eds.)
2. Gazi, V., Fidan, B., Hanay, Y.S., Koksal, M.Ä.: Turkish Journal of Electrical Engineering & Computer Sciences 15, 149 (2007)
3. Schmickl, T., Moslinger, C., Crailsheim, K.: Swarm Robotics, pp. 144–157 (2007), http://dx.doi.org/10.1007/978-3-540-71541-2_10
4. Gravagne, I.A., Marks, R.J.: IEEE Transactions on Systems, Man, and Cybernetics, Part B 37(2), 471 (2007)
5. Martinez, S., Cortes, J., Bullo, F.: IEEE Control Systems Magazine 27(4), 75 (2007) doi:10. 1109/MCS.2007.384124, http://dx.doi.org/10.1109/MCS.2007.384124
6. Balch, T., Hybinette, M.: Proceedings of IEEE International Conference on Robotics and Automation, ICRA 2000, vol. 1, p. 73 (2000), http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=844042
7. Batalin, M.A., Sukhatme, G.S.: Proc. of 6th International Symposium on Distributed Autonomous Robotic Systems, pp. 373–382 (2002)

8. Cassinis, R., Bianco, G., Cavagnini, A., Ransenigo, P.: Third European Workshop on Advanced Mobile Robots (Eurobot 1999), pp. 211–218 (1999)
9. Passino, K.: IEEE Control Systems Magazine 22(3), 52 (2002)
10. Batalin, M.A., Sukhatme, G.S.: IEEE International Conference on Robotics and Automation, ICRA 2003, pp. 2714–2719 (2003)
11. Rekleitis, I., Lee-shue, V., New, A.P., Choset, H.: Limited Communication, Multi-robot Team Based Coverage. In: ICRA 2004, pp. 3462–3468 (2004)
12. Soysal, O., Sahin, E.: A macroscopic model for self-organized aggregation in swarm robotic systems (2007), http://dx.doi.org/10.1007/978-3-540-71541-2_3
13. Luke, S., Cioffi-Revilla, C., Panait, L., Sullivan, K.: Proceedings of the 2004 SwarmFest Workshop (2004)

Analyzing Large-Scale Crowd Simulations for Building Evacuation*

Carlos García-Cabrera, Pedro Morillo, and Juan M. Orduña

Abstract. Animated virtual crowds have been used last years for analyzing human factors in scenarios where masses of people gather. A typical example is building evacuation in case of fire. Scalability still remains as an open issue for these multi-agent systems applications. In this paper, we use a scalable architecture to simulate a large-scale version of a virtual crowd in a building evacuation. From the social point of view, the results provided by the large-scale version of the crowd add new and crucial information about the agents behavior, emphasizing the need for a small amount of trained leaders in order to save lives. From the system point of view, the results show that the trend of avatars towards crowding in some areas highly increases the computation time for the agents hosted in some client computers. Therefore, this trend should be taken into account when designing large-scale evacuation simulations.

1 Introduction

The simulation of virtual crowds have been used last years for analyzing human factors in scenarios where masses of people gather, such as sporting events, transportation centers, buildings and concerts[12, 3, 1]. Usually, the motion of crowds and other flock-like groups is modeled as interacting particles that display different behaviors in 2D/3D virtual scenes [11, 2]. Beyond physically based simulations, agent-based crowd models aim to capture the nature of a crowd as a collection of individuals, each of which can have their own goals, knowledge and behaviors [10]. When virtual crowds are used for analyzing human factors in certain scenarios,

Carlos García-Cabrera, Pedro Morillo, and Juan M. Orduña
Departamento de Informática, Universidad de Valencia, Spain
e-mail: {pedro.morillo,juan.orduna}@uv.es

* This work has been jointly supported by the Spanish MEC and the European Commission FEDER funds under grants Consolider-Ingenio 2010 CSD2006-00046 and TIN2009-14475-C04-04.

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 591–598.
springerlink.com

local motion driven by Helbing's model [4] should be used. The requirements due to 3D graphics, motion and behavioral models add a huge workload to the computer supporting the crowd simulation.

In order to decrease the computational workload of simulations, crowds have been divided in different levels. For example, the ViCrowd system [9] divides the simulation at the level of crowds, groups, and individuals. Modern variants of these simulations use continuum dynamics to reach interactive simulation speeds for thousands of characters [12]. Although these approaches can display very populated and interactive scenes, their usability for analyzing human factors is questionable, since the higher-level behaviors are not based on individual behavior.

Also, there have been efforts to provide efficient and autonomous behaviors to crowd simulations [6, 8]. However, they are based on a centralized system architecture, and they can only control tens of autonomous agents with different skills (pedestrians with navigation and/or social behaviors for urban/evacuation contexts). The scalability of the provided results still remains as an open issue, due to the complexity of the relationships among different agents.

In this paper, we propose the use of a previously proposed system architecture [7] for simulating a large-scale version of the virtual crowd that analyzes human factors in building evacuation [8]. The purpose is on the one hand to test if the results are similar when the crowd size increases. In this sense, the results show a small probability of survival of those agents not achieving to evacuate the building in a short term, emphasizing the need for a small amount of trained leaders. On the other hand, the purpose is to analyze the performance of the underlying computer system, in order to determine the system requirements for large-scale simulations. In this sense, the results show that the trend of avatars towards crowding in some areas highly increases the computation time for the agents hosted in some client computers. As a result, the synchronous scheme followed in the simulation of building evacuation [8] is not suitable for large-scale simulations.

The rest of the paper is organized as follows: Section 2 briefly describes the implementation of the crowd simulation. Section 3 shows the the crowd behavior shown in the building evacuation, and also the performance evaluation of the simulation system. Finally, Section 4 shows some conclusions and future work to be done.

2 Architecture and Implementation of Building Evacuation

In order to implement a large-scale version of the behavioral model for building evacuations [8], different modifications have been made to the distributed architecture for crowd simulations [7]. This architecture controls how the existing agents can share information about the 3D virtual scene. We briefly describe this architecture here, in order to make this paper self-contained. It consists of a distributed computer system for supporting the software architecture shown in Figure 1. This software architecture is composed by two elements: the action server (AS) and the client processes (CP). The AS is devoted to execute the crowd actions, and it is hosted on a single computer, while each CP handles a subset of the existing agents and it is hosted on a single computer. Agents are implemented as threads of the CP for reducing the communication cost. Each thread manages the perception of the environment and the reasoning about the next action. Since reasoning formalisms can involve a high computational cost, each client process is hosted on a different computer, in such a way that the system can have a different number of client processes, depending on the number of agents in the system. In this way, this organization takes advantage of the underlying distributed hardware. This scheme allows the correct simulation of tens of thousands of autonomous agents at interactive rates.

The first modification consists of decoupling the different system functionalities in different layers. Concretely, we have implemented the low-level communication protocols, the mechanisms of inter-agent communications, and the functionality of generating new behaviors as different and independent software layers. The first layer, consisting of the low-level communication protocols, is based on the same model (based on BSD sockets) shown in the previous architecture for crowd simulations [7]. However, we have added a new kind of agents with new communication mechanisms, in order to extend the functionality of generating new behaviors. The second modification consists of the cells exclusively containing now information about which room they belong to. Agents only use the grid for knowing in which room they are currently located. With this information, they compute their own path to the exit. In this way, the system can dynamically react to both the creation and removal of exits during the evacuation. Additionally, each agent has a small memory that allows it to save which rooms have been already visited. These changes allow agents to make movements that obey a pre-defined behavior, depending on the kind of agents (trained leaders, non-trained leaders, normal agents, etc.). Once the path to be followed is computed, a force system based on the Helbing's model is used for moving.

Another modification of the previous architecture consists of the kind of information exchanged. In the previous architecture, each agent (a thread of a CP hosted

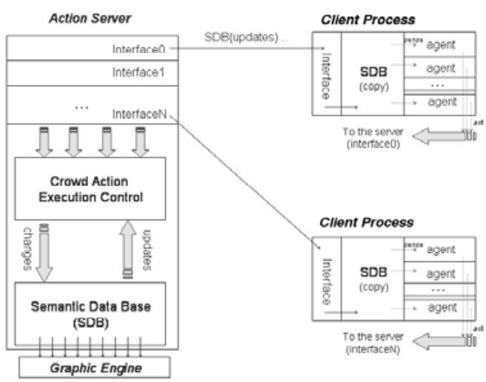


Fig. 1 Software architecture for large-scale building evacuation

in a given computer) exclusively sent to the server the new location (the point of the virtual world where it wanted to move). In the modified architecture, each agent not only sends its new desired location, but it also sends messages to other agents. These messages can be either destined to another agents (asking them to perform some actions) or messages describing the visited rooms.

Additionally, in order to provide this system with the same working scheme used for building evacuations [8], normal agents should send their requests to the server, completing the following cycle in each iteration: first, the agent should compute the path to follow, the Helbing forces, and the graphic updates for the graphical interface. If these computations are performed in less than 250 milliseconds (the agent cycle), then the agent (the thread) should sleep until this time has expired. At that moment, it wakes up and sends a request to the server for checking the computed movement. The server computes the answer in real-time, and it sends back this answer to the agent as soon as it is ready. When the answer arrives to the agent, then the cycle starts again. It should be noticed that, unlike the previous architecture for crowd simulations [7], there is no a server cycle (it computes the answer to each request as soon as it arrives), but an agent cycle. The reason for establishing an agent cycle is to provide all the agents with the same speed of movement. The reason for choosing 250 milliseconds as the cycle period is that it is the longer response time that human users perceive as interactive in DVE systems [5]. This period is the same used for building evacuations [8].

The goal for of all the agents is to leave the virtual scene. Regarding the agent functionality, we have implemented the same kind of agents shown in the behavioral model: normal agents, untrained leaders and trained leaders [8]. When the normal agents and untrained leaders see a hazard, they explore the maze (following a depth-first scheme in order to avoid cycles). The trained leaders know the correct path to the exits. Finally, the untrained leaders can exchange information with other agents about the state of the maze. Whenever two or more agents meet in a room, they share two pieces of information: locations of some of the hazards that are blocking paths, and parts of the building that have been fully explored by other agents and found to have no accessible exit (passed along by previous communications). The communication is local to a room, so agents exchange only relevant information about neighboring rooms, as for example, do not go through that door (there is fire), do not go in that direction (there is no exit), or follow me.

3 Performance Evaluation

In this section, we evaluate a large-scale version of the behavioral model for building evacuation [8]. Instead of a population ranging from 20 to 200 agents, we use a minimum of 400 agents and a maximum of 800 agents. However, due to space limitations we only show some representative results, corresponding to a population size of 400 agents. The results for other sizes were very similar.

The building evacuation consists of a structured square virtual world (a kind of maze) with a uniform initial distributions of the avatars (agents). That is, the ratio number of agents/room is constant. We have considered three different scenarios (maps). The first map contains two exits (in two opposite sides of the square) and two hazards that are quite close to the existing exits. The second map contains a random distribution of hazards within the maze. Finally, the third map is identical to the first map, but changing the exits to the opposite sides of the square with respect to the first map. Also, the two hazards are located close to the exits.

The first part of the performance evaluation has been the behavioral model. Figure 2 shows the performance (in terms of number of evacuated agents) of the different options for communication strategies and for a population of 400 agents. This figure shows that when communication among agents exists, then the performance is improved. Thus, when ninety simulations steps have been performed, around 360 agents (90% of the population) have been already evacuated if communication among agents exists, and only 270 agents (67.5% of the population) have been evacuated if no communications are exchanged. However, the rest of the population takes a similar number of steps to be completely evacuated, regardless of both the communication strategy and the size of the remaining population.

These results differ from the ones shown for 200 agents [8], where the simulation with communication finished (the population evacuated the world) in about half of the time that it took the non-communication case to finish. Similar results were obtained for greater populations and different maps. These results indicate that for large-scale evacuations, the communication strategy determines the portion of the population that can be evacuated on a short-term period, but it has not a noticeable effect on the long term performance. The reason is that the remaining population after the first simulation steps cannot exploit the knowledge provided by the communications. That is, the remaining agents are normal agents that do not communicate each other. The leaders manage to evacuate the building in the first steps of the simulation, due to their ability to share information.

Figure 3 shows the performance (in terms of the number of simulation steps in order to evacuate the building) for different simulations with different percentages of trained leaders. This figure shows that the only simulation where the population is fully evacuated in 120 simulation steps is the one with a 100% of trained leaders. The rest of simulations do not finish the evacuation after 510 simulation steps. In

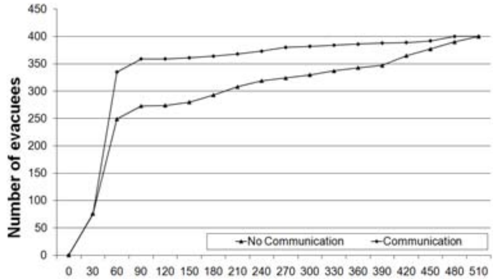


Fig. 2 Performance for different communication strategies

this sense, the performance is similar to the one shown in the previous figure. The performance provided by the different options considered greatly differs in the short term, but they tend to converge in the long term. Thus, for 120 simulation steps the biggest differences are shown between 0% and 25 % of trained leaders, with 270 and 350 agents evacuated, respectively.

If we compare these results with the ones provided by the behavioral model for building evacuation [8], we can see that they are proportional. However, the large-scale results (Figure 3) show that those agents that could not evacuate the building in a short term require a long time to reach their goal, with a low probability of survival. These results emphasize the need for a small percentage of trained leaders in building evacuations in order to save lives.

Additionally, we have analyzed the system performance, in order to characterize the system requirements for large-scale simulations. Since the underlying platform is a distributed computer system, the most important performance measures are latency and throughput. Figures 4 a) and b) show the system performance (in terms of average system responses and computation times, respectively) measured in the simulation of building evacuation for 400 agents. This figures contain three plots for the different maps considered. They show on the X-axis the simulation step, and they show on the Y-axis the average system response time or average computation time, respectively. The former one includes from the starting of the agent cycle to the instant when the server response arrives. The latter one measures the average computation time that agents require to complete the first part of the agent cycle (computing the path to follow, the Helbing forces, and the graphic updates for the graphical interface). Both Figures show the average values for one of the computers hosting either normal agents or non-trained leaders.

Figure 4 a) shows that the plots for the three maps show similar shapes and values. They range from 252 milliseconds to 260 milliseconds, and they alternate periods of peak values with periods of low latencies. The reason for these behaviors is the simultaneous waking-up of the agents (threads) in the same computer when the agent cycle finishes. Figure 4 a) also shows different simulation length for the different maps considered. While the simulation steps required for map 1 and map 2 are similar (both of them require around 350 simulation steps), map 3 requires

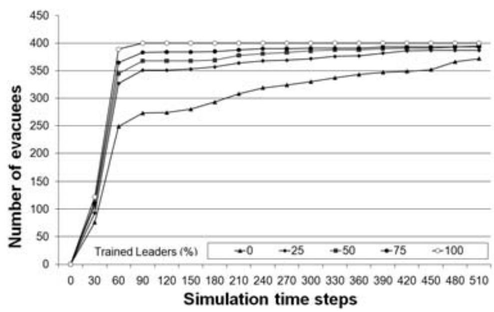


Fig. 3 Evacuation time for different numbers of trained leaders

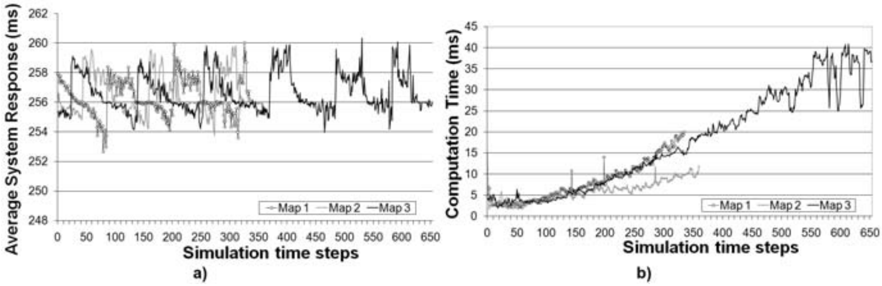


Fig. 4 a) average latencies and b) computation times for normal agents

around 650 simulation steps. It is due to the fact that there is a longer distance to the exits in map 3.

On other hand, figure 4 b) shows that, regardless of the considered map, the average computation time required for the first part of the agent cycle has a parabolic shape, continuously increasing from the start to the end of the simulation. Thus, the average computation time reaches the highest values for the map whose simulation lasts more time (map 3). The reason for this behavior is that the hazards are located close to the two existing exits in all the maps. Thus, agents tend to crowd in the rooms near the exits along time. As a result, the time required for computing the Helbing's model (where the forces produced by the surrounding agents should be taken into account) increases as so does the agent density. Although the computation times shown in Figure 4 a) are still far from the agent cycle, if the simulation lasts more time then computation times could reach the agent cycle, providing an unacceptable system performance.

Additionally, we have measured the same performance metrics in the computers hosting trained leaders, in order to analyze the system behavior for all the kinds of agents. Although they are not shown here due to space limitations, the computation time linearly decreases as the simulation proceeds, showing that the computers hosting trained leaders do not tend to saturation in large scale simulations.

4 Conclusions

In this paper, we have used a previously proposed system architecture to simulate a large-scale version of a virtual crowd that analyzes human factors in building evacuation. From the social pont of view, the results show that those agents that cannot be evacuated in a short term need a very long time to be evacuated (thus with a low probability of survival), regardless of the percentage of trained leaders. These results emphasize the need for a small amount of trained leaders for managing building evacuation.

Additionally, we have analyzed the system performance, in order to establish the system requirements for large-scale simulations. The results show that due to the trend of avatars to crowd in some areas, the computing time highly increases for some client computers as so does the population size. Since this trend is not bounded and the cycle period is constant, the computation times can exceed the cycle period for large-scale population sizes. The reason for this behavior is the saturation of the client computers. This trend should be taken into account when designing larger-scale evacuation simulations, and less normal agents or non-trained leaders should be hosted in each computer, in order to avoid system saturation.

References

1. Exodus, the evacuation model for the safety industry, http://fseg.gre.ac.uk/exodus/
2. Frank, T., Bernert, K., Pachler, K.: Dynamic load balancing for lagrangian particle tracking algorithms on mimd cluster computers. In: PARCO - International Conference on Parallel Computing (2001)
3. Galea, E.R., Sharp, G., Lawrence, P.J.: Investigating the representation of merging behavior at the floor–stair interface in computer simulations of multi-floor building evacuations. Journal of Fire Protection Engineering 18, 291–316 (2008)
4. Helbing, D., Farkas, I., Vicsek, T.: Simulating dynamical features of escape panic. Nature 407(6803), 487–490 (2000), http://www.citebase.org/abstract?id=oai:arXiv.org:cond-mat/0009448
5. Henderson, T., Bhatti, S.: Networked games: a qos-sensitive application for qos-insensitive users? In: Proceedings of the ACM SIGCOMM 2003, pp. 141–147. ACM Press/ACM SIGCOMM, New York (2003)
6. Iglesias, A., Luengo, F.: New goal selection scheme for behavioral animation of intelligent virtual agents. IEICE Transactions on Information and Systems E88-D(5), 865–871 (2005)
7. Lozano, M., Morillo, P., Orduña, J.M., Cavero, V., Vigueras, G.: A new system architecture for crowd simulation. J. Netw. Comput. Appl. 32(2), 474–482 (2009), http://dx.doi.org/10.1016/j.jnca.2008.02.011
8. Pelechano, N., Badler, N.I.: Modeling crowd and trained leader behavior during building evacuation. IEEE Comput. Graph. Appl. 26(6), 80–86 (2006), http://dx.doi.org/10.1109/MCG.2006.133
9. Raupp, S., Thalmann, D.: Hierarchical model for real time simulation of virtual human crowds. IEEE Transactions on Visualization and Computer Graphics 7(2), 152–164 (2001)
10. Reynolds, C.W.: Flocks, herds and schools: A distributed behavioral model. In: SIGGRAPH 1987: Proceedings of the 14th annual conference on Computer graphics and interactive techniques, pp. 25–34. ACM, New York (1987), http://doi.acm.org/10.1145/37401.37406
11. Sims, K.: Particle animation and rendering using data parallel computation. In: SIGGRAPH 1990: Proceedings of the 17th annual conference on Computer graphics and interactive techniques, pp. 405–413. ACM, New York (1990), http://doi.acm.org/10.1145/97879.97923
12. Treuille, A., Cooper, S., Popovic, Z.: Continuum crowds. In: SIGGRAPH 2006: ACM SIGGRAPH 2006 Papers, pp. 1160–1168. ACM, New York (2006)

Agent Collaboration Framework

Juan C. González-Moreno, Francisco J. Rodríguez-Martínez,
Rubén Romero-González, and Cástor Sánchez-Chao

Abstract. The philosophy of software development has changed significantly in recent years. Build quality software more quickly is imperative in todays market, specially when it has to meet the demands of consumers. Not long ago a development team could consist of only one member, while today this is unthinkable, since the number of members in the development teams increases with the complexity of the actual issues. Collaborative programming or pair programming is one of the more relevant areas at present in providing efficient solutions to the working groups management, in such a way that their impact on the software development process is indisputable. The main purpose of this document is to gather the most relevant aspects of the collaborative models in a collaborative framework based on the use of software agents. In practice, the proposed framework is designed in a generic manner, allowing its extension to other environments and different domains.

1 Introduction

Current society lives in a world of continuous change. A consequence of this change is the birth of the new technologies of the information and the appearance of the society of the knowledge,who provide a substantial improvement in quality of citizens life. As an example of this improvement, nowadays a person can communicate with another individual from almost anywhere in the world.

The increase and globalization in the scale of software projects has resulted in that the developments must be carried out through decentralization of the team. In this way is not necessary to work all together in an office in order to communicate changes, modifications, solutions, ... needed in the development process of any application. Thanks to the Internet and the variety of tools available for work in group, workers can make high quality developments using videoconferencing systems, blogs, chats, social networks, ... This new tools represent the new pillar

Juan C. González-Moreno · Francisco J. Rodríguez-Martínez ·
Rubén Romero-González · Cástor Sánchez-Chao
University of Vigo (Dpto. de Informática), Ed. Politécnico, Campus As Lagoas,
Ourense, 32004, Spain
e-mail: {jcmoreno,franjrm}@uvigo.es, rrglez@gmail.com,
cschao@cschao.org

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 599–606.
springerlink.com

for the development of the next generation applications, but at present their integration with traditional tools for software development is virtually non-existent.

1.1 Lacks of Collaborative CASE Tools

Nowadays CASE tools, show lacks in three important aspects related with collaborative tools: perception or awareness of group, communication, coordination and group memory. A geographically distributed developer team have usually problems of perception, due to the fact that in their development tools have not been integrated the collaborative services needed to know about what kind of activities have been doing the rest of their partners. Also the media are depleted or absent. Moreover the working memory of the group,(informal communication, notes on an item, corrections and so one) is forgotten bringing about a bad understanding of software requirements or just software design deficiencies.

2 Introduction to a Model

This work propose a framework which provides a theoretical solution to implement a modern collaborative platform. It allows the integration of different software tools for giving the needed collaborative support nowadays.

The proposal is based in the use of software agents to support communication between independent modules. The election has been done considering their distribution capacity, attendance, and also because the representation of the entities that are part of the software life cycle are totally covered.

A service that must be considered from a functional standpoint, is the maintaining of the history of actions that have been carried out during development (group memory). Thus, team members could move back or forward through all the changes made on the development. This work proposes an hybrid solution between local and global concept, to be able to carry out tasks locally and then integrate them globally or vice versa. A local actions register let developers work when the collaborative platform is not accessible. Thus upgrades could be performed synchronously or asynchronously. The final way will depend of which communication is available.

3 Architecture

As it shows on the figure1, the solution adopted has taken into account the basic features for a collaborative platform(perception or awareness of group communication, coordination and group memory).

The architecture presents a system based on two layers of communication. One part in local machine and other remote. The local part consist of two agents which will be responsible for establishing communications with the remote platform and keep a copy of the group memory or local history. On the other hand the remote platform will host players decision (Broker) and reporting (General History).

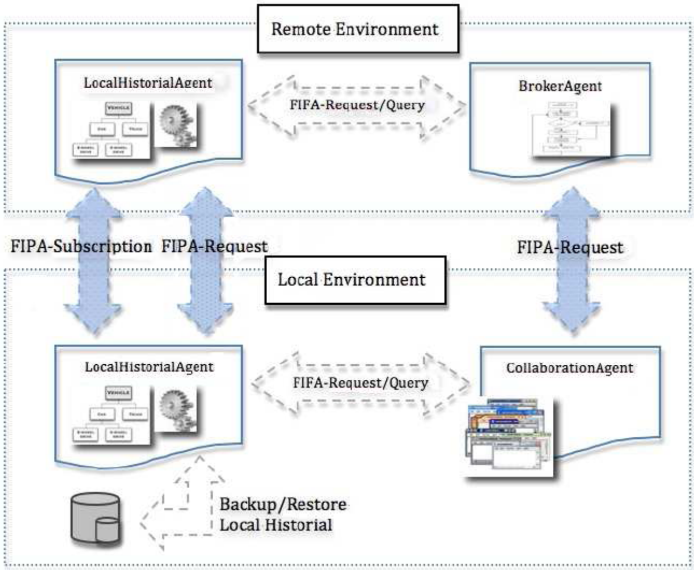


Fig. 1 Collaborative Architecture Model

3.1 Agents Integration

Every software project is developed by a team and every team have chain of command. Thought about it, it was considered that agents could coexist at different levels of integration, what implies to extend the concept of agent in a hierarchical system, to represent the different people that make up the team (project managers, analysts, designers, programmers).

Collaboration Agent will be responsible for representing the different entities that the project has, ie the development team. It will be supported by a graphical interface that will be responsible for forwarding the events that occur between the stakeholder and the machine. To summarize, possible actions the agent may perform are the following:

- Verify the existence of a broker agent that supports the decisions of the collaborative platform. If it dont exist, the agent must launch one.
- Order collaborative requests for Broker agent.
- Launch the local agent history, which be responsible for maintaining backups of the projects in which it participates, as also keep an updated copy of the work group.
- Provide the user with an interface as a gateway to the collaborative platform.
- Contact the local agent history, allowing the user to know the current state of development.

Local agent history must support the following features:

- Provide version control functions for projects in development
- Maintain a subscription updates against the general agent history so that it keep
- a copy of the collaborative development permanently updated.
- Provide a communication system between collaboration Agent and itself.

Broker Agent will be responsible of taking decisions related with actions to be done on projects that are offering in a collaborative way. To do it, its necessary to integrate a decision algorithm, by which it will decide whether the action requested by a stakeholder may be undertaken or not.

Finally, **general history agent** must provide mechanisms for different actors in local history can gets a copy of current history, as well as a real-time update, in case changes occur in the project.

3.2. Group Concurrency

Faced with a domain of these characteristics, it is always the question of how to resolve conflicts of concurrent modification. Therefore, we must consider the different options that are available to solve such problems.

Thinking about the types of development team members, one would think the concept of hierarchy, but moving it to concepts of agents, because as its expected, each agent will correspond to a person. As a result, actions to change the system state would be established in order of weight within the project. To wit: Project Manager, analysts, designers and programmers. However, it must take into account cases such as the fact that two equal-range designers, wishing to undertake the same type of action on the same entity.

Design strategies for this type of problem are quite different, here are some possible solutions.

- *Solution 1.* You can proceed by -race condition-, taking into account only the first which notify the intention to perform some action on a particular entity and rejecting all other requests for it, while changes are made.
- *Solution 2.* Accept all requests for action by the agents in the same entity. This would involve designing a fusion system of solutions provided. Such solutions usually have a fairly high order of complexity and which can hardly ever cover all the possibilities for action.
- *Solution 3.* Thinking of a resolution strategy based on composition of objects in CASE modeling tools, models, diagrams and metamodels, are made more basic entities. Taking it account, it could make a half-concurrent solution, it means, by applying the concept of -monitor-on components. In this way could maintain the level of exclusivity to a pre-specified atomicity. This strategy is considered a partial solution because of the implicit nature of the -monitor-.

3.2.1 Group Plans

In the previous section it talked about different solutions to concurrency problems between entities or objects. Each one of these solutions is perfectly valid if its restricted to a particular problem domain. A collaborative tool is usually enclosed within a very well defined frame, so that of the development of a framework has been thought that the best way to solve the problem is that the same user to write their own concurrency strategy. That is why i has developed a set of API's that allow the use with any solution.

This point has been a constant focus of working, cos decide which would be the strategy and tools to provide the user so that it could migrate its application had too much changing requirements, depending on the subject of the application domain.

3.3 Ontologies

The use of ontologies in the system to be developed is essential, since it will provide all support to store all the knowledge base resulting from the extrapolation of data models and diagrams, it means, the meta-modeling. Moreover, being an ontology in JADE consists of concepts, predicates and actions, it will enrich the sets of actions that may be carried out on the ontology, for example, relationships between entities, deletion, creation and modification of objects, and so one.

Using this structure raised by the Framework of TILAB, JADE, it has praised the idea of matching directly modeled objects from a CASE tool with it. The correspondence would be the following:

- **Predicates:** It represents the enriched portion of the system about questions/queries, it means, providing knowledge of the state of the ontology and access to it.
- **Action:** By achieving action, may alter the state of the ontology. That is, the use of actions related to the entities and relationships CRUDs, or purely administrative.
- **Concepts:** The concepts will establish a hierarchy of knowledge of the domain model. It will serve as currency for transactions carried out by actions or predicates.

4 Test Case

For the understanding of the use of framework built, it has been thought to use an existing CASE tool that provides support for collaborative work. For this reason we have chosen the INGENIAS[6]. As discussed in previous sections, it has built an ontology that will serve as a representation of the meta-modeling CASE which is performed using this tool. It is also envisaged to provide this ontology of actions to build and relate entities, the cornerstone of any diagram or model.

Actions, predicates and entities, these and if proportionate with the framework, have already been introduced in previous sections, so do not go into details here. Only mention that if it thinks of the concept of creating an determined entity, will be necessary to introduce a previous action to broker agent to satisfy the said petition and thus can be constructed such a role as shown in the diagram or model treat.

In a second phase, it has to decide a possible hierarchy of application usage as a function of the role that can occupy each of the development team members. It has been thought of: *project manager, analyst, designer, programmer*. To develop the group plan it had taken into account the possible actions on the same entity requested by equal roles and also applications for different roles. Figure 2, illustrates the plan implemented.

As it can see, it have been considered safeguarding times on a given entity, and control over property ownership. This kind of approach fully covers any event that can be raised to within the domain. However, thanks to the extensibility of the framework, nesting or use of another plan could be just as beneficial.

To test all implemented, it have been used to add the appropriate adapter logic to the CASE tool and thus can communicate with their new collaborative system. On the other side, it has developed a tool to facilitate the purification of the integration of the framework, this is a ”self-generated interface” which allows you to build any type of entity, action or predicate, and test it against the developed system. In this way you can check the operation of the plan implemented, which is well supported ontology to treat the problem or the use of history also, among others.

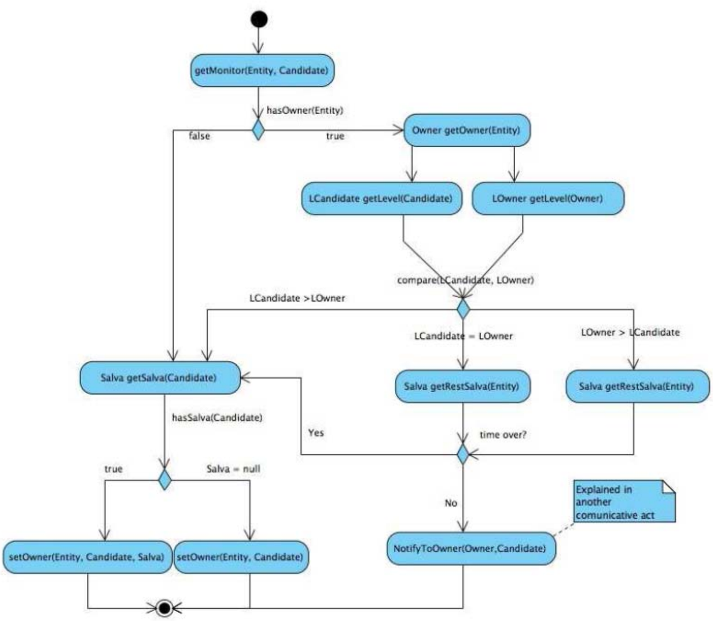


Fig. 2 Group planning template

5 Conclusions and Future Works

As you can see, the problem domain is quite extensive to treat. Focussing on the basis of the problem, the major aspects which helps in a collaborative develop work, that is built into the architecture allows keep a differentiation between local and global level regarding the project. Providing capacity for development back in the previous restore points, dynamic updates by synchronizing the actions of each history, and further remote platform independence to develop, since actions can be recorded in local history then update the overall project. Another distinguishing factor of this architecture is the intrinsic property for defining different group plans development plan. Give the user the ability to write their own concurrency strategies provide a degree of genericity and adaptability of the architecture, providing opportunities for application in any domain in which you want to apply collaborative development.

These two characteristics provide an extraordinary grade of operability in collaborative development environment, and today, represent a new development in the field of this type of applications.

As its presented in this work, it has developed a framework, still in alpha version, but serves as the basis for extending the tools that have no collaborative support. As future work aims to take account of issues such as high availability or improve support agent architecture, since the api that serves as the basis for the framework has algorithmic cost. It has also considered the possibility of migrating the architecture to different languages for further integration with other tools in the industry, so you can increase the scalability of the project. Following this line, one might think to keep core development based on a single language while the remaining components could be coupled with a different and use a language for communication, such as the one proposed by FIPA.

References

1. de Araujo, R.M., de Sousa Dias, M., Borges, M.R.S.: A framework for the classification of computer supported collaborative design approaches. In: CRIWG, pp. 91–100 (1997)
2. Lozano, M.D., Gallud, J.A., Martínez, A., Penichet, V.M.R.: Propuesta de diseño de un editor uml colaborativo basado en web. XV Jornadas de Ingeniería del Software y Bases de Datos (2006)
3. Dave, C.: Groupware, Workflow and Intranets. Reengineering the Enterprise with Collaborative Software (1998)
4. Graesser, A., Franklin, S.: Is it an agent, or just a program?: A taxonomy for autonomous agents. In: Jennings, N.R., Wooldridge, M.J., Müller, J.P. (eds.) ECAI-WS 1996 and ATAL 1996. LNCS, vol. 1193. Springer, Heidelberg (1997)
5. Canseco, V., Gerónimo, G.: Breve introducción a los sistemas colaborativos: Groupware & workflow (1998)
6. Gómez-Sanz, J.J., Fuentes-Fernández, R., Pavón, J., García-Magariño, I.: Ingenias development kit: a visual multi-agent system development environment (best academic demo of AAMAS 2008), Estoril Portugal, May 12-16, pp. 1675–1676 (2008)

7. Jennings, M., Wooldridge, N.: Agent technology - foundations, applications, and markets. Springer-UNICOM (1998)
8. Maes, P.: Modeling adaptive autonomous agents. Artificial Life Journal 1(1&2), 135–162 (1994)
9. Ortega, M., Martínez, M.I.: Aprendizaje colaborativo a través de guías móviles (2005)
10. Noriega, P., Rodríguez-Aguilar, J.A., Sierra, C.: Aplicaciones de tecnologia de agentes en comercio electrónico: conceptos, arquitectura y aplicaciones, pp. 231–241. Pearson-Prentice Hall, London (2005)
11. Nwana, H.: Software agents: An overview. knowledge engineering review. Technical report, pp. 1–40. Cambridge University Press, Cambridge (1996)
12. Ortega, M., Velázquez Iturbide, J.A., Paredes, M., Fernández, I.: Escritura colaborativa y pdas: una propuesta de aprendizaje basada en resolución de problemas (2003)
13. Andres Felipe Moreno Sagreso. Tesis. PhD thesis, Pontificia Universidad Catolica De Chile, Escuela De Ingenieria (1997)
14. Tony, B., Savarimuthu, R., Purvis, M.: A collaborative multi-agent based workflow system (2004)
15. XunWang, J., Tang, M.X.: A multi-agent framework for collaborative product design (2006)

Some Issues and Extensions of JADE to Cope with Multi-agent Operation in the Context of Ambient Intelligence

A. Paz-Lopez, G. Varela, S. Vazquez-Rodriguez, J.A. Becerra, and R.J. Duro

Abstract. Ambient Intelligence (AmI) can be taken as one of the principal test beds for the integration of Systems, Humans and Cybernetics. Multi-agent based Ambient Intelligence solutions have proliferated in the last few years in order to cope with the intrinsically distributed and complex interaction based nature of the problem. Different multi-agent platforms have been used in this regard, being JADE one of the standard solutions adopted due the variety of services it provides and to its FIPA compliance. Nevertheless, JADE was designed as a general Multi-agent platform and not for the particular demands of AmI systems. Consequently, its behavior in this realm is not as optimal as it could be. In this paper, we analyze some of the problems JADE presents for its application to AmI, especially in terms of communications, and describe some of the extensions we have developed in order to solve them. The resulting system exhibits enhanced communication capabilities, promotes the division of tasks into decoupled components and makes component reutilization easy. This platform constitutes an environment with tools for the development and deployment of AmI applications.

Keywords: Multi-agent systems, Ambient Intelligence, JADE.

1 Introduction

Ambient Intelligence is a multidisciplinary paradigm that takes inspiration from different fields like ubiquitous computing, artificial intelligence or biomedical sciences [1]. One of its main goals is to make technology disappear into the background, allowing the user to take control of its environment in a natural and completely transparent way. In fact, the idea is to make the environment proactive so that the user may have this control without even being aware of it. Many authors

---

A. Paz-Lopez · G. Varela · S. Vazquez-Rodriguez · J.A. Becerra · R.J. Duro
Integrated Group for Engineering Research, University of A Coruna
Mendizabal s/n, 15403, Ferrol, Spain
www.gii.udc.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 607–614.
springerlink.com

have worked on applying artificial intelligence based approaches to achieve this end in contexts like smart homes, elderly and health care or leisure activities, among others [2, 3].

Several problems arise when implementing such systems, especially due to the large number of sensors and distributed infrastructure they require to perform even the simplest tasks and consequently, to the ever more complex and abundant communications that must be handled as the systems scale up. These may be classified into hardware organization and integration related problems and software modularization and scaling related problems. Different and stimulating solutions have been proposed in terms of whole architectures from two points of view. On one hand, numerous solutions have adopted a service oriented approach like the PERSONA project [3], the AMIGO project [4] or [5]. On the other, some groups have resorted to agent based middleware for AmI such as in the case of the AIRE Project [6], the CHIL project [7] or our own HI^3 project, structured as an agent based layered architecture [8, 9].

Multi-Agent System (MAS) based platforms, when adequately designed, are a natural way to distribute intelligence and provide for component mobility, autonomy, scalability and fault tolerance [10]. In this line, many authors have resorted to tools and platforms that were developed for MAS but not necessarily for AmI [11, 12]. Thus, even though they simplify the generation and management of the MAS, many of the functionalities and requirements that arise in AmI have to be dealt with in an ad hoc manner. This is the case of JADE [13], a widely used MAS platform. It provides basic capabilities like inter-agent distributed communications, simple agent mobility, basic support for agent discovery and FIPA [14] compatibility. However, to achieve some of the requirements and be able to easily implement high level and AmI related features, JADE must be extended in certain regards. Some authors have already introduced some extensions aimed at their specific problems. A clear instance of this is the work by Moraitis and col. [11] with their Im@gine IT project [15] where they extended the FIPA architecture through the introduction of new agent types. However, they did not deal with some of the infrastructure or support elements that would be required for all agents to perform their tasks in a more efficient and trustworthy manner. These are the types of extensions we will consider here, they are not agents, but rather services provided by the platform for the agents to use as required such as improvements in the inter-agent communications capabilities, facilities to manage multi-agent conversations, introduction of a publish/subscribe communication paradigm or improved multi-agent model with support for declarative definition of components. All of this within a multi-agent system execution container with distributed management tools and development utilities to facilitate the implementation and deployment of new applications. Thus, the paper is organized as follows. Section 2 deals with the different extensions introduced on the communications. Next, in section 3, a declarative component model for AmI MAS systems is presented along with its related tools. Finally some conclusions are extracted.

2 Improving Agent Communication

Even though JADE provides good basic capabilities for inter-agent communications there is a need to introduce features that would help to adapt them to the special characteristics of AmI environments. Here we will describe three areas where extensions have been created: Inter-agent message exchange, multi-agent conversations and a publish/subscribe communications system.

In terms of Inter-agent Message Exchange, JADE provides a basic system for asynchronous message emission and reception that invites developers to code agents communications mixed with control logic in an ad-hoc manner. Due to this fact it is expensive to develop high level communications capabilities like agents with the ability to automatically know how to interact with other agents. One way to alleviate this problem could be to upgrade agent capabilities with a mechanism to encourage the decoupling between communications and control logic code.

Thus, we start by introducing the concept of Message Processor. A Message Processor is the most basic way of receiving messages in this extended platform. It is a behaviour that is associated to one or more message templates (they specify characteristics a message must fulfill) and is specialized on processing received messages. When a message is received that fulfills some message template, its associated Message Processor is woken up and the message is forwarded to it for processing. Message Processor elements will be in charge of applying any required preprocessing to the messages, decoupling it from the control logic to make it reusable, and finally, deliver the message to an adequate control behavior.

The message templates associated to each Message Processor completely specify the communications interface of an agent in any instant of time (Message Processors and templates can be added or removed dynamically). They thus provide a straightforward procedure to decouple communications code from the agent's functional logic and, when combined with the declarative agent model that will be presented later, provide basic support for declarative communications.

On top of the Message Processor concept, a priority based scheduling system for message reception was introduced. It provides developers with a mechanism to prioritize some agent interaction capabilities over others in order to focus the agent's attention on its most critical tasks.

Regarding Multi-Agent conversations, the JADE agent platform provides support for some predefined agent communications (even 1-to-N communications) based on the interaction protocols proposed by the FIPA standard. However, in a complex and broad environment like AmI, where there will be a large number of heterogeneous agents from multiple vendors, with complex behaviours and their own goals, it seems difficult to establish every complex interaction in a predefined way, especially when one of the goals of AmI is to build systems that autonomously evolve and adapt to the environment and its users.

One way to cope with those complex interactions could be to organize agents constituting societies and to provide them with mechanisms to participate in conversations with multiple agents through generic models that abstract them from the management logic of the interaction. Thus, the concept of conversation was introduced in the extended platform as a way to support multi-agent (N-to-M)

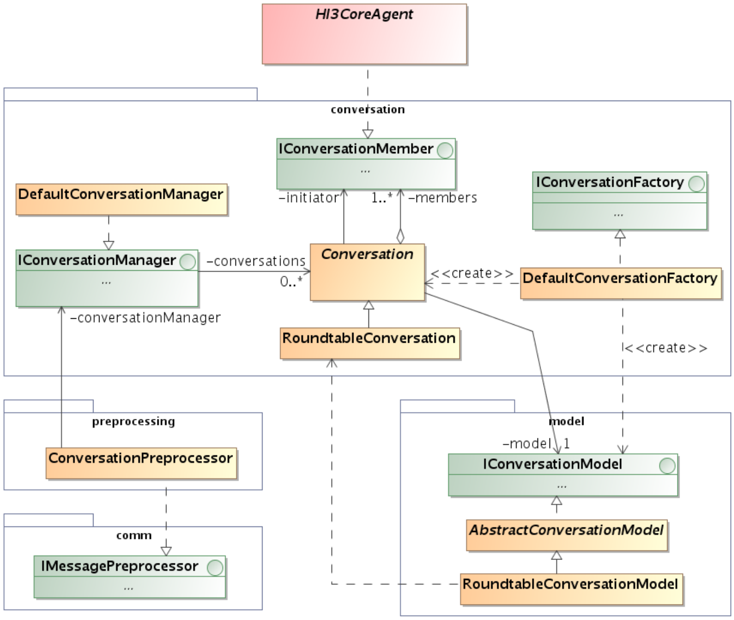


Fig. 1 Class diagram of the conversation management subsystem

interactions. Every agent has the integrated capability of being a member of a conversation. Therefore, (see Fig. 1) an agent is inherently capable of creating new conversations, of inviting new agents to a conversation, of sending/receiving messages within a conversation and of being informed about agents that enter or leave a conversation. As interactions between multiple agents can become really complex, the system is designed to easily support multiple types of conversations through the concept of Conversation Model.

Fig. 1 shows a simplified class diagram of the conversation subsystem design. It shows how conversations are controlled by a Conversation Manager that tracks messages of every known conversation by using an adequate Conversation Model implementation. These models contain the logic to drive a particular type of conversation, including how to invite new members to a conversation, how to leave a conversation or how to notify members of message reception.

In addition to the previously described communications models, we have added an extension for publish/subscribe communications. In this model, agents that wish to send some kind of information publish it as events, whereas agents interested in receiving certain types of events subscribe to those events.

A publish/subscribe model is a loosely coupled communications mechanism in which a publisher of an event is not necessarily aware of the recipients interested in that event or even of their existence. This communications model fits perfectly with typical interactions present in AmI environments like notifications about state

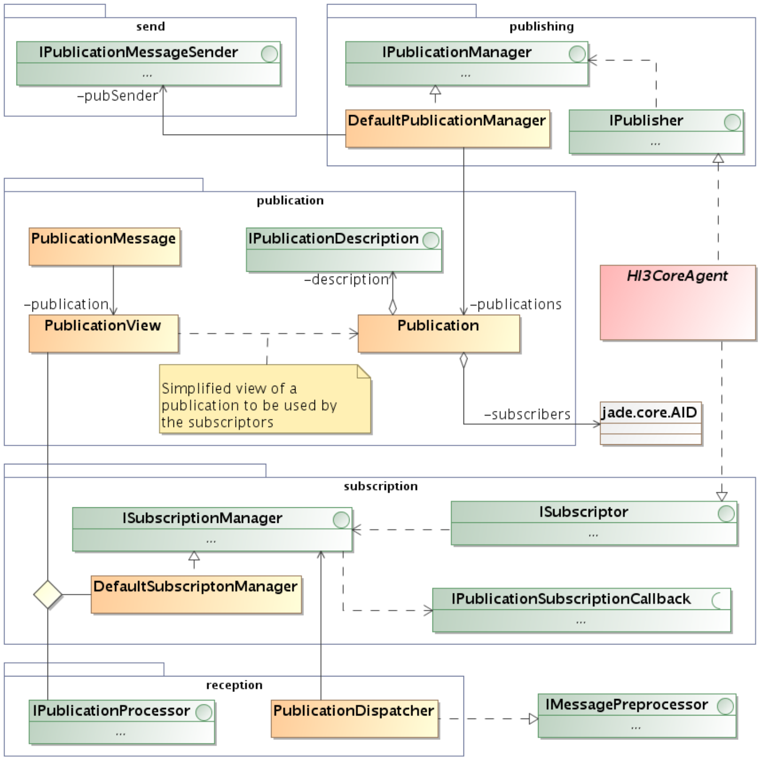


Fig. 2 Simplified class diagram of the publish/subscribe subsystem

changes in sensors, access to information about the lifecycle of other components or the platform itself, or sharing data between components.

As JADE does not provide a communications model of this nature, we have developed a publish/subscribe system on top of the communications extensions previously described. Within this system, agents inherently have the ability to create publications, publish content on those publications and subscribe to publications from other agents.

A simplified class diagram of the publish/subscribe subsystem is shown In Fig. 2. The management of the publications and subscriptions is decoupled from the agents by the use of a publication and a subscription manager that encapsulates the logic and data required to receive publications and to send content to the subscribed components. These managers rely on the message exchange subsystem of the extended platform described previously. Furthermore, developers can, in a declarative manner, define and describe the set of publications for a given agent. Then, at run-time, other heterogeneous agents can discover publications and dynamically subscribe and unsubscribe to them.

3 Multi-agent Declarative Model

As mentioned before, AmI systems tend to be really large and complex systems but, at the same time, they usually share a lot of common functionalities. One possible option to alleviate this complexity is to promote the division of the system into highly decoupled components. This is one of the main purposes of our extended platform from a conceptual point of view and it is supported in the software platform by what we call the Multi-agent declarative model.

The Multi-agent declarative model is, on one hand, a model to describe a multi-agent system and its components and, on the other, a set of tools to declaratively define these components. As a model, it brings together all the different elements the platform leaves in the hands of the developers to build new applications. The top level concept of the platform is the multi-agent system (MAS). The platform is prepared to support various types of MAS with different high level characteristics but, in its basic form, they are a collection of instances of different types of agents that work together to perform a task.

Directly below the MAS concept is the agent-type, which specifies the components that define an agent in the platform, providing a way to declaratively build new agent-types by reusing preexisting components and mixing them with new ones. Agent-types are defined according to a set of concepts that also have independent and declarative definitions:

- Behaviours. They contain the implementation of the agent's functionality.
- Message Processors. Define the communications interface of an agent.
- Publications. Specify the publications that the agent offers for subscription.
- Arguments. Allow the parameterization of agents at instantiation time.
- Functionalities. Describe the functionalities that an agent wants to publicly advertise to other agents. When an agent instance is launched, the platform automatically announces its functionalities on the platform agent registry.

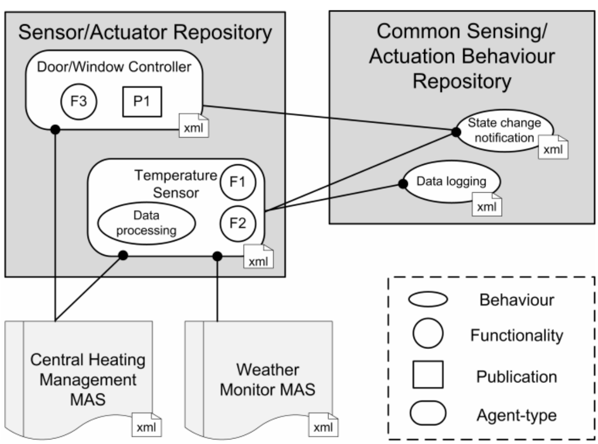


Fig. 3 Declarative model usage and component reutilization example

Furthermore, Agent-types support inheritance. An Agent-type can inherit the components specified for its parent Agent-type, so that the instances of the child Agent-type will have the behaviours, message processors, etc. of the parent.

This conceptual model is backed by a set of tools that provide declarative support to define and use these concepts. These tools are mainly made up of an XML based language for component description, and utilities to create and manage these description files.

Every major component has its own definition in an XML file, making them independent and reusable. To support that reusability, an URL based system was designed for component referencing. This provides a flexible and transparent way to distribute and reference reusable components.

An example of using the Multi-agent declarative model is shown in Fig. 3. It is easy to see how every component has its own independent definition, except those that are internal to other components, like the "Data processing" behavior of the "Temperature Sensor". It also shows the use of component repositories, from which new components can be created by the aggregation of preexisting ones.

4 Conclusions

JADE has been revealed as a suitable general purpose MAS platform, but, based on our experience in AmI, we think that it requires of a set of extensions that enhance its functionalities and adapts it to the problems it will really find. In this paper we have argued for and described some of these enhancements.

A new subsystem for message reception and processing was implemented here with support to decouple communications code from functional logic. This subsystem provides the basics to build systems in which components can specify their communications interface to others. In addition, two high level communications models that provide integrated support for typical interactions of the AmI world, like dynamically changing interactions between multiple components, through the conversation concept, or highly decoupled event-based communications through a publish/subscribe model have been created. Finally, a component conceptual model has been introduced on top of JADE agents and behaviours. It decouples a system into a set of multiple independent components that can be shared by multiple systems. This model is supported by tools like an XML language to describe the components and a URL system for transparent component referencing and aggregation. As a whole, it provides a standardized way to divide a system into components and to reutilize logic between systems.

Summarizing, to adapt JADE to the realm of AmI, some of its functionalities must be extended and adapted to this context. This paper is a first step in this direction presenting the some JADE extensions that have been implemented and tested as the base of the AmI architecture developed within the HI^3 project [9].

Acknowledgments. This work was partially funded by the Ministerio de Educación y Ciencia of Spain through project DEP2006-56158-C03-02.

References

1. Remagnino, P., Foresti, G.L.: Ambient Intelligence: A New Multidisciplinary Paradigm. IEEE Trans. SMC- Part A 35(1), 1–6 (2005)
2. Brdiczka, O., Crowley, J.L., Reignier, P.: Learning situation models in a smart home. IEEE Trans. SMC - Part B 39(1), 56–63 (2009)
3. PERSONA, PERceptive Spaces promoting iNdependent Aging, `http://www.aal-persona.org/deliverables.php`
4. AMIGO, Ambient intelligence for the networked home environment, `http://www.hitech-projects.com/euprojects/amigo/software.htm`
5. Wu, C.-L., Liao, C.-F., Fu, L.-C.: Service-Oriented Smart-Home Architecture Based on OSGi and Mobile Agent Technology. IEEE Trans. SMC 37(2), 193–205 (2007)
6. Peters, S., Look, G., Quigley, K., Shrobe, H., Gajos, K.: Hyperglue: Designing High-Level Agent Communication for Distributed Applications. In: Proceedings of AAMAS (2003)
7. Soldatos, J., Dimakis, N., Stamatis, K., Polymenakos, L.: A breadboard architecture for pervasive context-aware services in smart spaces: middleware components and prototype applications. Personal and Ubiquitous Computing 11(3), 193–212 (2007)
8. Varela, G., Paz-Lopez, A., Vazquez-Rodriguez, S., Duro, R.J.: HI3 Project: Desing and Implementation of the Lower Level Layers. In: Proc. IEEE VECIMS, pp. 1–6 (2007)
9. Paz-Lopez, A., Varela, G., Monroy, J., Vazquez-Rodriguez, S., Duro, R.J.: HI3 Project: General Purpose Ambient Intelligence Architecture. In: Proc. AITAmI 2008, pp. 77–81 (2008)
10. O'Hare, G.M.P., O'Grady, M.J., Keegan, S., O'Kane, D., Tynan, R., Marsh, D.: Intelligent Agile Agents: Active Enablers for Ambient Intelligence. In: Proc. of the Ambient Intelligence for Scientific Discovery Conference (2004)
11. Spanoudakis, N.I., Moraitis, P.: Agent Based Architecture in an Ambient Intelligence Context. In: EUMAS (2006)
12. Doctor, F., Hagras, H., Callaghan, V.: A Fuzzy Embedded Agent-Based Approach for Realizing Ambient Intelligence in Intelligent Inhabited Environments. IEEE Trans. SMC- Part A 35(1), 55–65 (2005)
13. JADE. Java Agent DEvelopment Framework, `http://jade.tilab.com`
14. FIPA Agent Com. Lang. Spec., `http://www.fipa.org/repository/aclspecs.html`
15. Moraitis, P., Petraki, E., Spanoudakis, N.: An Agent-Based System for Infomobility Services. In: EUMAS 2005, Brussels, Belgium (2005)

P300-Based Brain-Computer Interface for Internet Browsing

José L. Sirvent, José M. Azorín, Eduardo Iáñez, Andrés Úbeda,
and Eduardo Fernández

Abstract. This paper describes the implementation of a Brain-Computer Interface (BCI) for controlling Internet browsing. The system uses electroencephalographic (EEG) signals to control the computer by evoked potentials through the P300 paradigm. This way, using visual stimulus, the user is able to control the Internet navigation via a virtual mouse and keyboard. The system has been developed under the BCI2000 platform. This paper also shows the experimental results obtained by different users.

Keywords: Brain Computer Interface, P300, Evoked potentials, Virtual keyboard.

1 Introduction

A Brain-Computer Interface is defined as the hardware and software that allow interacting with a computer without the need of any muscular movement [1]. This technology is specially designed for people who suffer from ALS paralysis or movement limitations. This is perhaps the only way they can interact or communicate with the environment.

For this interaction, various control paradigms have been investigated: invasive (mainly in animal research) where by implanting small intracranial sensors it is possible to obtain signal patterns of a small group of neurons [2], and non-invasive, where external sensors are placed on the patient's scalp [3]. The latter avoid the risks of surgery and do not produce any injury to the brain tissue. Among non-invasive techniques, we can highlight two control paradigms. In the *Spontaneous Mental Activity paradigm*, the frequency patterns captured on the motor cortex change when motor mental task are performed [4]. This paradigm

José L. Sirvent · José M. Azorín · Eduardo Iáñez · Andrés Úbeda ·
Eduardo Fernández
Biomedical Neuroengineering Group
Universidad Miguel Hernández de Elche
Avda. de la Universidad s/n, 03202, Elche (Alicante), Spain
e-mail: {jsirvent,jm.azorin,eianez}@umh.es,
a.ubeda.cast@gmail.com, e.fernandez@umh.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 615–622.
springerlink.com

could be used for one-dimensional movement of a cursor [5]. The *Event Related Potential* or P300 paradigm is based on obtaining a characteristic signal pattern over the parietal cortex produced approximately 300 milliseconds after producing visual, tactile or auditory stimulus [6] (See Fig.1. right).

The control paradigm using evoked potentials was first proposed by *Farwell and Donchin* [6]. They developed a speller using a 6x6 character matrix, where rows and columns flash randomly. Until now this system has proved itself to be the safest and most reliable because of its high hit rate and good classification. This paradigm has been applied in navigation systems on wheelchairs [7], or Internet browsers such as *Nessi* [8][9] (where the links are associated to visual stimulus).

This paper describes a Brain-Computer Interface that allows controlling a computer. This interface substitutes the physical keyboard and mouse input by virtual devices which are controlled by brain activity. Therefore, it is not necessary to use specific software to perform tasks as Internet browsing, sending e-mails or controlling Windows (amongst others). The user is able to control the keyboard and the mouse by using evoked potentials (P300). Among others BCI for controlling computers, the main advantage of this system is that the user is able to control all the computer functions without any special interface, so all Windows functions can be done.

The paper is organised as follows: The hardware and software of the BCI is explained in Section 2. Section 3 describes the appearance and workflow of the system. The tests and results obtained by the users while testing the interface is explained in Section 4. Finally, in Section 5 the main conclusions are summarized.

2 Brain Computer Protocol

2.1 EEG Hardware

The main hardware used in the Brain-Computer Interface is based on the gTec commercial devices, consisting of: gUSBAMP amplifier, which utilises 16 input channels where 16 g.EEGelectrode sensors will be connected. For the placement of the sensors on the user scalp, a special EEG cap which comprises of 64 positions has been used. The computer and the amplifier are connected by USB. Fig.1. (left) shows the equipment used.

The electrodes have been placed, according to previous studies [10] [11], in the next positions following the 10/20 standard: Fz, C3, Cz, C4, Cp3, Cp4, P5, P3, Pz, P4, P6, Po7, Po3, Po4, Po8 and Oz, ground in Fpz and reference to the right ear lobe.

2.2 Feature Extraction

The EEG signal has been amplified and digitalized with a sample frequency of 256 Hz using 8 bits per sample. Later a notch filter has been applied; reject-band

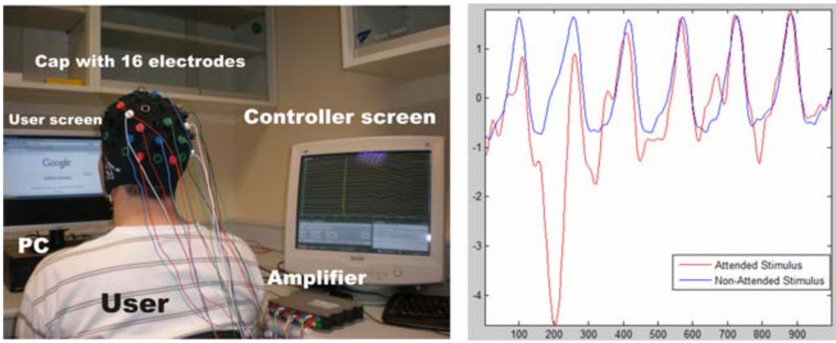


Fig. 1 Left: Image of the equipment used and the laboratory environment. Right: Signal produced by attended and non-attended stimulus in sensor Po8.

between 48 and 52 Hz, to avoid induced electromagnetic interferences produced by the 50 Hz power line. Finally the signal has been filtered again with a band-pass filter between 0.1 and 30 Hz, this frequency band is suitable for the P300 signal.

The feature extraction process followed is a temporal filtering of the signal produced by a visual stimulus. This filter only considers the first 600 milliseconds after the stimulus to study. This new signal has 153 samples per channel and will be the input signal for the classifier.

For sampling and signal processing the BCI2000 software has been used [11]. This software, installed in a computer (Windows XP, Intel Core 2 2x2.66Ghz 1Ghz RAM), is used for the signal registration, processing and classification.

2.3 Classification

The classifier will be responsible for distinguishing between the waveform produced by an attended or a non-attended stimulus. The *Stepwise Linear Discriminant Analysis* (SWLDA) has been used as classifier [13]. This algorithm is widely used for the classification of evoked potentials. Previous studies prove its high performance for this kind of signals [11].

This classifier performs a linear transformation, a matrix multiplication of a classification matrix with the input signal matrix (the output of the temporal filter). The classifier looks for the optimal discriminant function adding features (in this case, consisting in channels and time elements) to a lineal equation, obtaining the maximal variability and optimal separation between two classes.

The classification matrix consists in a matrix of coefficients obtained from the training signals of each volunteer. P300 classifier (a BCI2000 contribution [11]) has been used to obtain this matrix. The configuration used is as follows: study time 0-600 ms, maximal features 60, Spatial filter RAW, channel set 1-16.

3 Implementation

3.1 Developed Software

A software application module has been developed to allow controlling a Windows-based computer (including Internet browsing). The application has various selection menus (with rows and columns flashing randomly) where the user has to fix their gaze on/stare at one option to choose the desired item. These menus consist of virtual devices which allow the same interaction with the computer as the real keyboard or mouse, so the selected options reproduce the same actions as a real device.

3.2 Graphical Interface

The graphical interface of the application must be used with a 1024x768 resolution and it will be placed in the bottom of the screen; the width fits the screen and is 250 pixels high, as is shown in Fig. 2. This way the user has 2/3 of the screen for Windows or Internet navigation.

Fig. 2 Graphical interface of the system: Virtual keyboard (left), Virtual mouse (center) and motion interval (right)

The selection matrixes are as follow: 11x4 for the virtual keyboard, 5x3 for the virtual mouse and 3x3 for the distance control. The keyboard has all the basic keys for introducing numbers and characters; moreover there are keys with n/f (no function) which are free for future modifications. In the mouse matrix there are 8 possible directions, right click, left click, double left click, scroll up and down, read mode (in which the system will ignore the produced stimulus) and change to keyboard. Finally in the distance matrix there are 9 options with different distances of movement for the cursor, from 2 to 800 pixels in different intervals. With these basic functions the system is able to control almost all the computer functions.

3.3 Execution Protocol

The execution protocol is the way the user controls the possibilities of the interfaces presented in Section 3.2. This succession of orders and commands are modeled on a very basic state machine, see Fig.3.

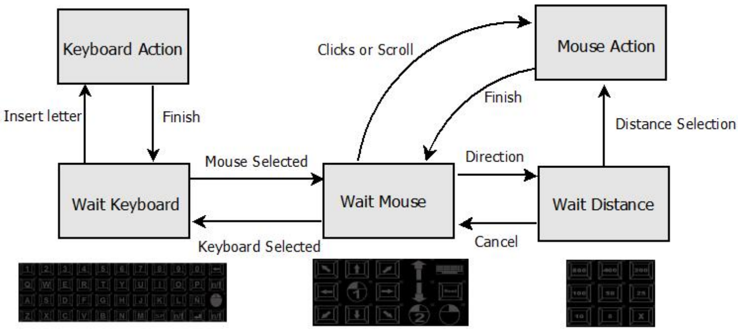


Fig. 3 States Machine of the system

The program execution starts by showing the keyboard (Fig.2 left) with all its options in the state *Wait Keyboard*. At that moment the computer will not execute any action in the input devices. When the user concentrates on one of the characters or options which are flickering on the screen, the system starts the features extraction of the signal and the classifier is responsible for recognizing the patterns and determining the desired order. When the desired character flashes 12 times (6 per row and 6 per column) the program executes the action associated to the election obtained from the classificatory algorithm. When a character is selected the system changes to state *Keyboard Action* where the computer inserts the desired letter. When the *change to mouse menu key* is selected the system will change to the next menu and it will be in *Wait-mouse State*. In this menu we have commands that can be executed with one single option (such as clicks and scroll), changing automatically from the system state to *Mouse Action*.When the desired action is executed. Although, if the selection is cursor movement, we have to specify the distance we want the cursor to move, or, in case of an error, we will have to cancel the selection. To do this, the third selection menu will appear and the system will be in *Waiting-Distance* state. After the distance selection (in pixels) is done, the system will execute the cursor movement in the *Mouse Action* state. Once the movements and actions related to mouse are finished, we can go back to the keyboard selecting the *Keyboard* command on the mouse menu, returning to the initial state.

4 Experimental Test

4.1 Participants

To test the system the participation of 5 healthy users between the ages of 24 and 33, without any gender restrictions, has been required. After informing the participants of the requirements and tests involved, the volunteers agreed and gave their consent to take part in the tests.

4.2 Protocol

The protocol applied to carry out the test was the same for all the users and consisted of the following phases:

1) Training: Training of the SWLDA algorithm is essential for a correct operation of the system. For a correct training it is necessary to do various training trials selecting certain known commands. The Copy Spelling function of BCI2000 has been used to carry out the sampling. The users undergo 2 sessions of training by selecting the commands associated to *COMPUTER* and *SCIENCE*. These words were selected because of their distribution across the keyboard, so the stimuli contributions were spread through the widest possible visual field. We selected the best trial to extract the SWLDA coefficients to train the application.

2) Writing: The users were asked to write in a text editor any particular word to test the correct operation of the classifier. This test was performed to check the hit rate.

3) Cursor: The user was asked to move the cursor to reach an objective on the screen, the time taken was measured. This way we were able to see how quickly the cursor can reach any point on the screen. The proposed tasks are: Task 1, reach a textbox and Task2, reach a square area of 100x100 pixels.

4) Google search: Finally, a complex task is proposed to the users. In this case the user has to perform two Google searches. Every volunteer will do the task moving the cursor to the Google textbox from the bottom left corner of the browser, doing click there and writing UMH and VR2 (one word in each trial), selecting the *Intro* character and doing click in the first link which appears in the Google list. Using the time measured to complete the task we will determine the system usability for complex tasks.

4.3 Experimental Results

All the tests have been done under the following conditions: Stimulus time 31.25ms, time between stimulus 125ms, random stimulus presentation.

For testing, the users have carried out each task presented in Section 4.2 twice, to be able compare the results and to see their variability. The results are shown in Table 1.

Table 1 Results obtained from the experimental test

Participant	Writing Test		Cursor Task		Cursor Task 2		Google Search	
	1st.	2nd.	1st.	2nd	1st.	2nd.	1st.	2nd.
User 1	42%	a	7:25	a	a	a	11:12	a
User 2	75%	100%	1:37	1:15	4:20	3:05	5:38	5:00
User 3	100%	100%	1:30	1:31	2:35	3:42	4:31	3:48
User 4	100%	100%	1:14	0:43	1:34	1:33	3:58	4:25
User 5	87.5%	100%	1:32	0:42	1:54	2:15	3:50	3:51
Global Average (b)	93%		1:12		2:25		4:20	

*a: The user did not do the test. *b: User 1 out of study.

As shown in Table 1 the necessary time for each task is relatively short, except for user 1 who admitted at the beginning of the session that he was extremely tired, which is the reason of his bad results (the evoked signal produced by attended and non-attended stimulus was almost the same, so, in his case, the classificatory algorithm could not discriminate both states).

The present system configuration provides the following selection speed: 3.5selections/minute for the keyboard and 6selections/minute for the mouse. In terms of accuracy, the hit rate average of the participants is 93%. We also noticed in the study of the volunteer signals that the difference between attended and non-attended is bigger in the zones P5, Po7, Po3, Po4, Po8 and P6. Moreover it is interesting to check that in some cases the accuracy and time for the tasks has improved in the second trial, this means that the users learned quickly to control the computer with this system.

5 Conclusions

A new BCI for Internet browsing has been presented and tested. In this BCI, the user is able to control the input interfaces of a computer using the P300 paradigm and the obtained results are satisfactory for both simple tasks as well as for complex tasks. The system has a high usability and does not have any interaction limitation with the computer. Moreover we agree on the effectiveness of the SWLDA for P300 classification.

In the future, this interface will be tested by a volunteer who suffers paralysis. In addition, this BCI will be modified to control a robotic arm using a similar configuration.

Acknowledgements. This research has been supported by grant DPI2008- 06875-C03-03 (Ministerio de Ciencia e Innovación) from the Spanish Government. We thank all the BCI2000 and subsequent applications, as P300Speller or Stimuli Presentation, developers.

References

1. Cabrera, A.F., do Nascimento, O.F., Farina, D., Dremstrup, K.: Brain-Computer Interfacing; How to Control Computers with Thoughts, Department of Health Science and Technology. Aalborg University
2. Velliste, M., Perel, S., Chance Spalding, M., Whitford, A.S., Schwartz, A.B.: Cortical control of a prosthetic arm for self-feeding. Nature, International Weekly Journal of Science (2008)
3. Wang, Y., Gao, X., Hong, B., Jia, C., Gao, S.: Brain-Computer Interfaces Based on Visual Evoked Potentials. IEEE Engineering in Medicine and Biology Magazine (September/October 2008)
4. Millán, J., del, R., Franzé, M., Mouriño, J., Cincotti, F., Babiloni, F.: Relevant EEG Features for the Classification of Spontaneous Motor-Related Tasks. Biological Cybernetics 86, 89–95 (2002)

5. Wolpaw, J.R., McFarland, D.J., Neat, G.W., Porneris, C.A.: An EEG-based brain-computer interface for cursor control. Electroencephalographic Clinical Neurophysiology 78(910) (2008)
6. Farwell, L.A., Donchin, E.: Talkin off the top of your head; Torward a mental Prosthesis utilizing event-related brain potentials. Electroenceph. Clin. Neurophysiol. 70, 510–523 (1998)
7. Iturrate, I., Antelis, J., Kubler, A., Minguez, J.: Non-Invasive Brain-Actuated Weelchair Based on a P300 Neurophysiological Protocol and Automated Navigation. In: ICRA (2009)
8. Mügler, E., Bensch, M., Halder, S., Rosenstiel, W., Bogdan, M., Birbaumer, N., Kubler, A.: Control of an Internet Browser Using the P300 Event Related Potential. International Journal of Bioelectromagnetism 10(1), 56–63 (2008)
9. Bensch, M., Karim, A.A., Mellinger, J., Hinterberger, T., Tangermann, M., Rosenstiel, W., Birdbaumer, N.: Nessi: An EEG-Controlled web Browser for Severely Paralyzed Patients. Computational Intelligence and Neuroscience 5 (2007)
10. Krusienski, D.J., Sellers, E.W., Cabestaing, F., Bayoudh, S., McFarland, D.J., Vaughan, T.M., Wolpaw, J.R.: A comparasion of classification tecniques for the P300 Speller. Journal of Neural Engineering 3, 299–305 (2006)
11. Mirghasemi, H., Fazel-Rezai, R.: Analysis of P300 Classifiers in Brain Computer Interface Speller. In: EBMS Annual International Conference 2006 (2006)
12. Schalk, G., McFarland, D.J., Hinterberger, T., Birbaumer, N., Wolpaw, J.R.: BCI2000: A General-Purpose Brain-Computer Interface (BCI) System. IEEE Transactions on Biomedical Engineering 51(6), 1034–1043 (2004)
13. Johnson, G.D., Krusienski, D.J.: Ensemble SWLDA Classifiers for the P300 Speller. Human-Computer Interaction. Novel Interaction Methods and Techniques, pp. 551–557 (2009)

Modeling Virtual Agent Behavior in a Computer Game to Be Used in a Real Enviroment

Catalina Roncancio, Jaime Gómez G-B, and Eduardo Zalama

Abstract. Our long term goal is to develop autonomous robotic systems that have the cognitive abilities of humans, including communication, coordination, adapting to novel situations, and learning through experience. Cognitive architectures as theory of the fixed mechanisms and structures that underlie human cognition are the actual mechanism of making a software implementations of a general theory of intelligence. The proposed system incorporates the hypothesis behind cognitive architectures like Soar to model our particular content, an autonomous character and its cognitive processes in normal working situations as hotel bellboy, and simulated in a virtual environment. Through this work, we proposed introduce game development as a test bed for our application.

Keywords: cognitive modeling, computer game, programing by demosntration.

1 Introduction

The growing interest on cognitive models have let the integration on different disciplines, like psychology, linguistics, anthropology, and artificial intelligence, to name just a few, that have characterized the cognitive systems as systems which exhibit adaptive, anticipatory, and purposive goal-directed behavior. We can find various paradigms of cognition, each taking a significantly different stance on the nature of cognition, what a cognitive system does, and how a cognitive system should be analyzed and synthesized [3, 4]. Each paradigm has a history of asking certain types of questions and accepting certain types of answers. And that, according to Allen Newell, is both an advantage and a problem. What we finally need is to work with unified theories of cognition (UTCs) [12] and this is what the cognitive architectures attempts to be [13]. To understand how any computational architecture

Catalina Roncancio · Jaime Gómez G-B · Eduardo Zalama
CARTIF, Parque Tecnologico Boecillo parc. 205 Valladolid, Spain
e-mail: {catron,jaigom,eduzal}@cartif.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 623–630.
springerlink.com

works, we need to used it to model some behavior [5] (an architecture by itself does nothing, it requires content to produce behavior). Lets consider the next scenario: Sacarino(also named AiE-Agent) is a bellboy robot that will work in a hotel environment. In the field of service robots, there has been an increase in the necessity of developing assistant robots capable of interacting with users and undertaking real life tasks. Notwithstanding the general consensus about the necessity of developing and exploiting the potential of service robots, it stills remains a manifold challenge to do so today. Our assistant bellboy robot for hotels will be designed and developed capable of providing the following services and tasks:

- Accompanying the guests to their rooms.
- Explaining the services available in the room and the hotel (meals, laundry, etc).
- Carrying food, drinks, equipment, newspapers to the rooms.
- Dialogging with guests in defined contexts, taking care of orders, providing useful information (tourist and meteorological information, news, etc.), and sending/receiving messages to/from the reception desk.

The proposed system incorporates game design and artificial intelligence to simulate autonomous characters and their cognitive processes in normal working situations. The combination of computer games with virtual agent systems gives the opportunity to offer virtual environments for improving training and decision-making with virtual autonomous agents in everyday surroundings. One important advantage to bring the real application in to a simulation is that we can exhibit a high degree of anthropomorphism [7] with highly expressive interfaces that are easily adjusted and personalized for each user at a fraction of the cost of a robotic interface. Our paper is structured as follows. Section 2 presents some related work. Section 3 Explains the details by designing a game within the XNA Game Studio. Section 4 describes our approach to modeling the bellboy game ,and shows how we used our model to generate code and section 5 discusses the benefits of our approach and concludes.

2 Related Works

A number of noteworthy architectures for behavioral animation of autonomous characters have been proposed. In many of these techniques, characters behave autonomously by choosing actions through a behavioral model: an executable model that defines how a character should react to its environment. Our approach does not model the behavior explicitly. The behavior emerges based on the interaction. The use of visual modeling environments is not new to the gaming industry but if its use as a test bed of real application. The main objective of developing such systems is be able to work in parallel with the real application and then yet anticipate the most probably simulated situations before getting to work with the robot. For example, the Soar architecture approach is also applying on computer games design [15]. Through this work, they investigate some designs that facilitate tractable reinforcement learning in symbolic agents developed using Soar architecture operating in a complex domain, Infinite Mario. A reinforcement Learning domain developed for Reinforcement Learning Competition 2009, as is a variant of Nintendo Super Mario.

Another related work is [10], where they proposed a system that incorporates virtual reality and artificial intelligence to simulate virtual autonomous characters and their cognitive processes in dangerous working situation on an industry. The cognitive agents are enriched with a planner for selecting actions according to goals, the environment and to the personal characteristics of the agents (time pressure, caution, tiredness, hunger). A work that comes close to our research has been done by Jrg Kienzle et al. [6] on modeling computer games, and Non-Player Characters, to reason about the behavior of a tank pilot for the EA Tank Wars competition, in which Computer Science students compete against each other by writing artificial intelligence (AI) components that control the movements of a tank.

3 Execution Platform

On this first stage, our challenge is to develop a virtual environment that will support the actions of the virtual agent represented by a non-player character in a computer game. This will let us develop a demonstrator where users can interact with our virtual robot service and obtain information about the hotel. The modeling of our robot service Sacarino in a game with appropriate abstraction level using an appropriate modeling language has many advantages: (i) Programming-by-demonstration interface. (ii) Writing consistent, re-usable and efficient AI code. (iii) Provides the user with a social interface that acts as a representative of the services the environment offers. And (iv) enhance human-machine interaction. The game is an interesting tool, as in the real application, it is requires an agent to reason and learn at several levels; from modeling sensory-motor primitives to path-planning and devising strategies to deal with various components of the environment. We propose the integration of Soar theory to define the agent's behavior with a virtual model of the environment (by XNA). The description of the system will be based on statecharts [2, 17], a combination of state diagrams and class diagrams that interpret the differents modules of the system.

3.1 General Game Structure

In the recently years the design of Computer Game becoming an interesting test bead for research in Artificial Intelligence and Machine Learning [14, 15]. Traditionally, the research has concentrated on learning in Board game. A good example of a board game that has seen many successful computerized implementations is Chess [11]. However, recently computer and video games (Real-time games) have received increased attention [8, 16] because they present challenges which are close to real world problems like that of the enormity of information in a highly self contained and circumscribed environment. Real-time games requires users controls one character (or a small number of characters), and plays within a game environment against a set of computer controlled characters (or in multi players games against characters controlled by other players). In such games, the term artificial intelligence is used to designate the algorithms that specify the behavior of computer-controlled game characters, often also called non-player characters (NPC). The ultimate goals is to make the NPCs own actions and reactions to game events seem as intelligent and natural as possible. The central logic for every game includes preparing the environment where the game will run, running the game in a loop until the game ending criteria is met, and cleaning up the environment [1]. The idea of having the main program logic running in a loop is crucial for a game, because the game needs to keep running whether or not it has user interaction. This doesn't happen with some commercial applications, which only do something in response to user input. Comparing XNA with the old way of developing game we can see that the Game project type provides us with a ready made basic game structure, so we can start by including our game-specific code, and focus in the main target (Intelligent behavior and improve interaction). To illustrate the power of our approach, we show in the following sections how we modeled the bellboy robot behavior in a game.

4 Modeling Game

In games or simulations, a character perceives the environment through his senses or sensors, and reacts to it through actions or actuators. For instance, our character might look for the presence of a person and if its seen subsequently decide to make an action. The basic architecture is described in Fig. 1. In the perception module we find the information of processing video and voice recognition. The Cognitive module based on Soar [9], performs the behavior planing separating memories for descriptions of the current situation and its long-term knowledge. The current situations, including data from sensors, active goals, and active operators is held in the working memory. The knowledge that exists independent of the current situation is held in the architectures long-term memory (LTM). LTM is not directly available, but must be searched to find what is relevant to the current situation of the bellboy. Soar distinguishes three different types of LTM: procedural, semantic, and episodic. Procedural knowledge is primarily responsible for controlling behavior and maps directly onto operator knowledge. Semantic and episodic knowledge usually come into play only when procedural knowledge is in some way incomplete or inadequate for the current situation. The Behavior System is the processing component that generates behavior out of the content that resides in the long-term and working memories. The purpose of the Behavior system is to select the next operator to apply. A goal directed behavior corresponds to movement in a problem space from the current state to a new state through the application of an operator to the current state.

4.1 Modeling the State of the Bellboy

The visual scene is divided into a two-dimensional [16 x 10] matrix of tiles. Each tile (element in the scene) can have one of the many values that can be used by the agent to determine it the corresponding tile in the scene is a obstacle (objects), or not, a user, or a receptionist, etc. For every visible user, the agent is provided with

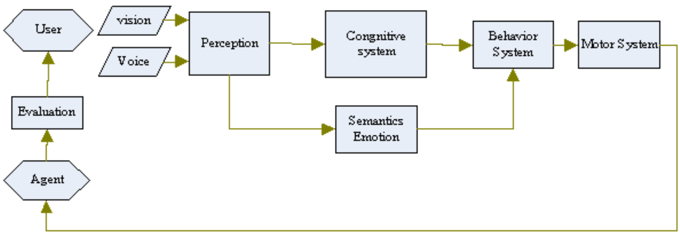


Fig. 1 Architecture

the type of the user, its current location, and its speed in both x and y direction. Once it determines the type of user, the characters initiates a dialog with it and the information to be interchange is store in the agent's memory. At the high level of abstraction, the bellboy has a given physical size, approximated by a bounding rectangle, with eyes (cameras), ears (stereo microphone), and arms. This set of sensors relay information about the state of the character and the surrounding environment to the AiE-Agent. The Chase component tell the AiE-Agent the position of the character in which direction the character is facing and what speed is going at. The mood component shows the current mood and translates them in their physical counterparts. Finally a battery indicator shows the current battery level of the bellboy, and a status indicator reports on the current interaction level.

4.2 Cognitive System

Its clear that if we want to see how Soar contributes to behavior then we need to explore the it in terms of some particular content [9]. Lets consider a simple scenario from the hotel Fig. 2. Sacarino stands around the hotel reception waiting for a guest arrival, and then initiates a dialog. First the guest came into the reception and makes the check in, and then the receptionist indicates to the bellboy, where to go (the guest's room). At that point the bellboy has to initiate a dialog with the user while they go to the room. The dialog is about hotel's information or services (laundry, room service, safe, staff, wake-up call etc). When they are near the room, the bellboy if its necessary indicates to the guest how to use the key card to open the door. Then the bellboy explains the hotel's mealtimes or other information. Finally the bellboy offers the user additional information, like, entertainment, touristic places, leisure and so on.

Just as the architecture is a theory about what is common to cognition, the content in any particular model is a theory about the knowledge the agent has that contributes to the behavior. For our AiE-Agent to act like a bellboy, we will have to give it many different kinds of knowledge. Before our model can dialog its first time, we must find some way to represent and process bellboy's knowledge in Soar. In Soar this structure provides a means for organizing knowledge as a sequence of decisions through a problem space. Some concrete examples of which are:

Fig. 2 Show the participants in our scenario in their usual locations and roles

- K1: Knowledge of the objects in the game. e.g. Lobby, front office, elevator, rooms, bar.
- K2: Knowledge of abstract events and particular episodes. e.g. What the user had mentioned in his previous interaction with the bellboy.
- K3: Knowledge of the rules of the game. e.g. Interaction with the bellboy.
- K4: Knowledge of objectives. e.g. Give information of the hotel's services, leisure, mealtimes and transport.
- K5: Knowledge of actions or methods for attaining objectives. e.g. Chase, Evade,Surround, tell jokes.
- K6: Knowledge of when to choose actions or methods. e.g. If the user don't need information, stop the dialog and go away.
- K7: Knowledge of the component physical actions. e.g. What to express (facial expressions) while dialogue

This list incorporates many different kinds of knowledge that our model must include: knowledge about things in the world (K1 and K2) and knowledge about abstract ideas (K3 and K4), knowledge about physical actions (K7) and knowledge about mental actions (K5), even knowledge about how to use the other kinds of knowledge (K6).

4.3 Behavior System

If we try to imagine (and draw) all the choices Sacarino might have to make during a game, given all the circumstances under which they might arise, we are quickly overwhelmed. Sacarino must make his decisions with respect to the situation at the moment. At the highest level of abstraction,as seen in Fig. 3, the AiE-Agent switches between different operating modes based on events. He starts in Exploring mode, and switches to Interaction mode once the Avatar position is known (and there is still enough battery). If at any point in time the mood status is negative or the interaction level is low, he switches to standby mode. Otherwise, Surrounding is the

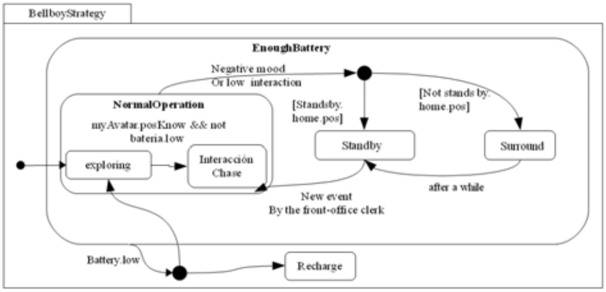


Fig. 3 The bellboy strategy

best strategy. In the event that the battery is low, if the location of the recharge station is known, the bellboy chooses to switch to recharge mode. Otherwise, it is best to continue Exploring, hoping to find a recharge station soon. When the charge is full, the bellboy switches back to whatever he was doing before he was interrupted. The mode changes are announced by sending corresponding events: when Exploring is entered, the explore event is sent, when Interaction is entered, interact event is sent, etc. The motor system is not explained in this work, this first phase is to model the correct behavior of a bellboy robot in this virtual environment.

5 Conclusions

Our decision of design a virtual agent is based on the modern conception of agent, while in the past software agents and robots have usually been seen as distinct artifacts of their respective domains, the modern conception is, in fact, to consider them as particular instances of the same notion of agent, an autonomous entity capable of reactive and pro-active behavior in the environment it inhabits. Our technique combines a form of cognitive modeling, based on the most development architecture in the literature, Soar, with emotions to influence decision making. In this paper we present a novel technique to produce intelligent behavior of an agent by interacting with the environment (simulated specific scenario) and the user. We model this interaction between the guest and a bellboy robot through a programming by-demonstration interface. Our technique produces virtual character behavior that can be quickly adopted when playing the game by non-expert users. Our method is a test bed of the ROBOTEL project a real-world application within hotel environment.

Acknowledgements. This research is based upon works partly supported by the Science and Innovation Spanish ministry (Pr. Nb.DPI2008-06738-C02-01/DPI)), jan-2009 to dec-2011 and CARTIF foundation.

References

1. LoBao, A., Evangelista, B., de Farias, J.A.L.: Beginning XNA 2.0 Game Programming: From Novice to Professional (2008)
2. Harel, D., Kugler, H.: The rhapsody semantics of statecharts (or, on the executable core of the UML), 325–354 (2004)
3. Vernon, D., Metta, G., Sandini, G.: A Survey of Artificial Cognitive Systems: Implications for the Autonomous Development of Mental Capabilities in Computational Agents. IEEE Transactions on evolutionary computation, special issue on autonomous mental development (2007)
4. Garzón, F.J.C.: Arquitecturas de la cognición, Mente = Cerebro + Medio (2008)
5. LehMan, J.F., Laird, J., Rosenbloom, P.: Agentel Introduction to Soar, an Architecture for Human Cognition: 2006 Update. University of Michigan (2006)
6. Kienzle, J., Denault, A., Vangheluwe, H.: Model-based Design of Computer-Controlled Game Character Behavior (2007)
7. Kopp, S., Gesellensetter, L., Krämer, N.C., Wachsmuth, I.: A conversational agent as museum guide – design and evaluation of a real-world application. In: Panayiotopoulos, T., Gratch, J., Aylett, R.S., Ballin, D., Olivier, P., Rist, T. (eds.) IVA 2005. LNCS (LNAI), vol. 3661, pp. 329–343. Springer, Heidelberg (2005)
8. Laird, J., van Lent, M.: Human-level AI's killler application: Interactive computer games. AI Magazine (2001)
9. Laird, J.: Extending the Soar Cognitive Architecture. In: Artificial General Intelligence Conference, Memphis, TN (2008)
10. Edward, L., Lourdeaux, D., Barthès, J.-P., Lenne, D., Burkhardt, J.-M.: Modelling Autonomous Virtual Agent Behaviours in a Virtual Environment for Risk. The International Journal of Virtual Reality (2008)
11. Newborn, M.: Deep blue's contribution to AI. Ann. Math. Artif. Intell. 28(1-4), 27–30 (2000)
12. Newell, A.: Unified Theories of Cognition. Harvard University Press, Cambridge (1990)
13. Langley, P., Laird, J.E., Rogers, S.: Cognitive architectures: Research issues and challenges (2006)
14. Ponsen, M., Spronck, P., Tuyls, K.: Towards Relational Hierarchical Reinforcement Learning in Computer Games. In: Proceedings of the 18th Benelux Conference on Artificial Intelligence, Belgium (2006)
15. Mohan, S., Laird, J.E.: Learning Play Mario. Technical Report CCA-TR-2009-03. Center for Cognitive Architecture. University of Michigan (2009)
16. Spronck, P., Ponsen, M., Sprinkhuizen-Kuyper, I., Postma, E.O.: Adaptive game AI with dynamic scripting and Machine Learning. Special Issue on Machine Learning in Games (2006)
17. UML Resource Page, http://www.uml.org (last access December 10, 2009)

A Legal View of Ambient Assisted Living Developments

J.P. Pedraza, M.A. Patricio, A. De Asís, and J.M. Molina

Abstract. In this paper, a legal approach to Ambient Assisted Living is considered. A general framework for context-aware applications is presented and a general view of legal principles to be considered in this framework. The analysis of a specific application of AAL, developed in a previous work, allows understanding these principles in a real development and the applicability for designing AAL applications.

Keywords: AAL, Context Applications, User Identification, Social Guarantees, Privacy and Human Rights.

1 Introduction

The concept of Ambient Intelligent (AmI) includes the contextual information but expand this concept to the ambient surrounding the people. So, electronic or digital part of the ambience (devices) will often need to act intelligently on behalf of people. It is also associated to a society based on unobtrusive, often invisible interactions amongst people and computer-based services taking place in a global computing environment. Context and context-awareness are central issues to ambient intelligence [1]. AmI has also been recognized as a promising approach to tackle the problems in the domain of Assisted Living [2]. Ambient Assisted Living (AAL) born as an initiative from the European Union to emphasize the importance of addressing the needs of the ageing European population, which is growing every year as [3]. The program intends to extend the time the elderly can live in their home environment by increasing the autonomy of people and assisting them in carrying out their daily activities. Several prototypes encompass the functionalities mentioned above: Rentto et al. [4], in the Wireless Wellness Monitor project, have developed a prototype of a smart home that integrates the context

J.P. Pedraza · A. De Asís
Public Law Department, Universidad Carlos III de Madrid, Colmenarejo, Spain
e-mail: {jpedraza@der-pu,aeasis@der-pu}.uc3m.es

M.A. Patricio · J.M. Molina
Computer Science Department, Universidad Carlos III de Madrid, Colmenarejo, Spain
e-mail: {mpatrici@inf,molina@ia}.uc3m.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 631–638.
springerlink.com

information from health monitoring devices and the information from the home appliances. Becker et al. [5] describe the amiCa project which supports monitoring of daily liquid and food intakes, location tracking and fall detection. The PAUL (Personal Assistant Unit for Living) system from University of Kaiserslautern [6] collects signals from motion detectors, wall switches or body signals, and interprets them to assist the user in his daily life but also to monitor his health condition and to safeguard him. There are also several approaches with a distributed architecture like AMADE [7] that integrates an alert management system as well as automated identification, location and movement control systems.

All these approaches are promising applications from an engineering point of view, but, no legal aspects are considered in the development. Clearly, an important point is the necessity to identify the users of these systems. Two different approaches could be considered, one approach is based in the cooperation of the user to be identified and another one is based in the non-cooperative environment (for example in surveillance applications). Biometric technology has legal implications because it has the potential to reveal much more about a person than just their identity. For instance, retina scans, and other methods, can reveal medical conditions. Thus biometric technology can be a potential threaten to privacy [8]. European and American judges [9] have categorized privacy as taking three distinct forms. These includes [10]: a) physical privacy or freedom from contact with other people; b) decisional privacy or the freedom of the individual to make private choices about the personal and intimate matters that affect her without undue government interference and c) informational privacy or freedom of individual to limit access to certain personal information about oneself. Obviously, biometrical technology is related with the a) and c) issues. Biometric identification, of course, is not a new technology. Introduced more than a century ago, fingerprint technology is perhaps the most common biometric identification technique. Thus the social risk [11] associated to this technology is not new. However, technological advances, among other factors [12], have increased the social risk associated to technique because: a) they have reduced the social tendency to reject its use; b) they have allowed their widespread use [13] and c) they have enabled to obtain more sensitive information on the subject.

In this work, authors review legal consideration in biometric identification to propose a set of legal principles on a general context aware framework. Finally a real application is studied from these principles.

2 Legal Consideration in Biometric User Identification

States and stakeholders should make further efforts to ensure that biometrical applications are monitored and the rights and freedoms of individuals are respected [14]. In particular, they should take into account, inter alia: the legal nature of relations (public or private) and the characteristics of the devices (ability to obtain sensitive information):

a) Private Relations (Private Users and Private Services) [15]. Because most biometric scanning will result from private sector activities where the user voluntarily gives up information, legal privacy concerns will usually be implicated to ensure informed consent and the transparency with the data subject. This is achieved providing them with the information about the systems and granting the right to access to personal data and, where appropriate, the right to have it deleted or rectified or blocked if they are inaccurate or have been unlawfully processed [16].

b) Public Relations (Private/Public Users and Public Services). In this context, the social guarantees, depends on the particular case and the results of legal test of the "balancing interests" [17][18]. There are common principles to "balancing interest" test: proportionality and reasonableness.

The principle of proportionality requires that measures implemented should be appropriate for attaining the objective pursued and must not go beyond what is necessary to achieve it. The reasonableness of a measure is therefore to be adjudged in the light of the nature and legal consequences of the relevant remedy and of the relevant rights and interests of all the persons concerned.

Also in this field, States shall ensure that appropriate procedures guaranteeing the dignity and privacy of the applicant, in particular, the protection of personal data. The States concerned shall closely monitor the implementation of the social guarantees, including: a) the general information on features and uses of systems; b) all the technical and organizational security measures required to protect personal data against accidental or unlawful destruction or accidental loss, alteration, unauthorised disclosure or access, and all other unlawful forms of processing the personal data; c) the collection and transmission of biometric identifiers; d) any processing of personal data must be lawful and fair to the individuals concerned; whereas, in particular, the data must be adequate, relevant and not excessive in relation to the purposes for which they are processed; whereas such purposes must be explicit and legitimate and must be determined at the time of collection of the data; whereas the purposes of processing further to collection shall not be incompatible with the purposes as they were originally specified; e) in all cases the level of security shall be adapted to the sensitive nature of the data; f) in general, the techniques taken to ensure compliance with data protection provisions and provide a mechanism for citizens to access, control, and verify their information. Society as a whole needs to be aware of the obligations and rights that are applicable in relation to the use of biometric applications. Therefore it makes sense to create a regulatory model for the collection, use and dissemination of biometric information. In that regard, there're several options like laissez faire approach, self-regulation, public regulation [19][20][21]. Under a laissez faire regime, no authority requires businesses to disclose their biometric policies to consumers. Therefore, it would be difficult for customers to comprehensively weigh the alternatives. The self regulation is not sufficient because entails one big drawback: the lack of enforcement. The last alternative deals with binding legislation with effective, proportionate and dissuasive sanctions for infringements.

3 Regulatory Model for AAL Developments

A generic framework of an AAL Application consists of three layers as shown in Figure 1. At the bottom of the Model is the Location/Monitor Layer, which is responsible for processing sensory information received by a collection of heterogeneous sensors into useful information for Context-Aware services. The set of sensors that monitor the activity of individuals are often organized in so-called sensor networks. With this sensory information, the Context Layer aims to answer the questions previously raised. Therefore, it is necessary to process and model information through the "Physical Context Manager" module. This module, in turn, is able to interact with the sensory layer in order to select certain preferences in the operation of the sensors. An AAL application should adapt its sensory information dynamically to the needs of users, taking into account a wide range of users and situations they may encounter. Through the "Logical Context Manager" module, the system is capable of adapting Context-Aware sensory information based on knowledge about their needs and characteristics, stored in what is called "Personal Profile". This profile or logical context can be obtained directly through inputs by the user preferences, or by interaction with the environment observed from the sensory system. With the merger of the logical and physical context information, the system is able to obtain the Context-Aware "User Model". The "User Model" plays a critical role in Context-Aware systems, since it embodies, on the one hand, the high-level semantic knowledge of actions of the user received from the sensory system; on the other, the user preferences, as well as its capabilities and limitations. Among these limitations, we may include information on their cognitive and sensorial level, or physical disabilities (for instance, elderly or handicapped people). Finally, once established "User Model", the Context Layer has a "Reasoning System" module capable of inferring and accommodating the needs of services to the final users in the field of a specific Context-Aware application.

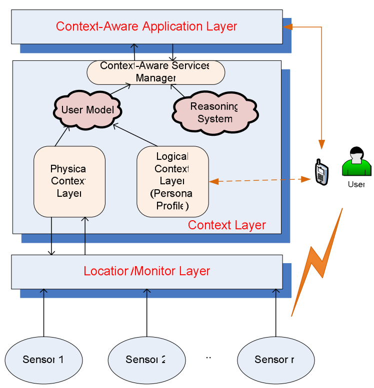


Fig. 1 Abstract Model of Context-Aware Applications

Identification and personalization are essential features of AAL services. The contextual framework needs a biometric scheme with the following features: (a) multibiometric: which combines several sources of biometric information (traits, sensors, etc.) with the aim of mitigating the inherent limitations of each source, obtaining a more reliable and accurate system; (b) highly transparent, highly accepted, and low intrusive, using biometric traits that can be acquired even without any cooperation of the user (e.g. face, voice) and well socially accepted (like the handwritten signature); (c) able of inferring human activity and analyzing user emotions, therefore significantly focused on services customization. These requirements affect directly to many legal aspects that should the considered before the development of industrial applications, to be used in the private sector or public sector. A generic legal framework of a Context-Aware Application should be composed by principles and fundamental rules, taking into account: (a) Central axiological elements: The protection of human dignity, fundamental rights and in particular the protection of personal data, are the key issues of regulatory model; (b) Principles: This regulatory model and a range of implementing measures needs to be adopted to complete the legal framework, should duly take into account some general principles. From our point of view, the general principles that should be taken into account could resume in the following ones:

1. Public objective driven vs. technology driven: the legal treatment for context-aware applications should not be ‘technology-driven’, in the sense that the almost limitless opportunities offered by new technologies should always be checked against relevant human rights protection principles and used only insofar as they comply with those principles.
2. Proportionality: requires that measures implemented should be appropriate for attaining the objective pursued and must not go beyond what is necessary to achieve it. The use of biometrics should not in principle be chosen if the objective can also be reached using other, less radical means.
3. Reasonability: reasonableness of a measure is therefore to be adjudged in the light of the nature and legal consequences of the relevant remedy and of the relevant rights and interests of all the persons concerned.
4. Data governance: is a useful principle that covers all legal, technical and organizational means by which organizations ensure full responsibility over the way in which data are handled, such as planning and control, use of sound technology, adequate training of staff, compliance audits, etc.
5. Human rights protection by design: human rights protection requirements should be an integral part of all system development and should not just be seen as a necessary condition for the legality of a system.
6. Best Available Techniques: shall mean the most effective and advanced stage in the development of activities and their methods of operation which indicate the practical suitability of particular techniques for providing in principle the basis for ITS applications and systems to be compliant with Human rights protection requirements.
7. Precautionary: where there is scientific uncertainty as to the existence or extent of risks to human rights, the institutions may take protective measures without having to wait until the reality and seriousness of those risks become fully apparent.
8. Technology neutrality: regulatory framework must be flexible enough to cover all techniques that may be used to provide context-aware applications.

4 An AAL Case of Study

Several AAL developments have been carried out in our laboratory, a complete description could be consulted in [22][23][24]. In these applications, the provisioning of the services occurs automatically in the Context Engine as the right context is found to each user: Role, Zone, Location, etc… For example, a grandmother sitting in a wheelchair with who's carrying a WiFi device and who usually take her medications every day, so the following rule is defined and discovered by the system:

```
Scenario I: Intelligent Home + Elderly + Taking Medication
Event part: When Rose Mary, the grandmother of the family,
            carrying a PDA is detected in the TV room,
Condition part: (and) it is the first time in the day,
Action part: (then) turn on the device, and send a MEDICATION'S
            ALERT.
```

The following rule is evaluated in order to offer the appropriate services to the elderly woman who is in the TV room. The intelligent home is able to know the location of each person at home (using cameras or wifi), identify each one (using cameras or wifi), correspond each mobile device with people who carry out, and apply context-rules to inform each user. In this simple example, some legal consideration should be done, following the principles of the proposed regulatory model (section 4):

1. Public objective driven vs. technology driven: the device could offer higher level functionalities in an automatic way but considering public goal and "the principle of the independence of will", the device should be configured in order to capture the information defined by the user.
2. Proportionality: the identification system does not need a personal recognition based on cameras only the identification of the device is necessary.
3. Reasonability: in this application the message send to the user could be turn off (other applications need to be always turn on, for example, in a hospital the message should send to medical assistance to be considered in any case).
4. Data governance: the whole system is under personal data privacy law.
5. Human rights protection by design: user should be able to configure the way in which the alarm is showed in order to avoid the publicity of the personal situation to other people at home.
6. Best Available Techniques: the designed devices should consider the minimum effort from the user and a low cost.
7. Precautionary: the technology involved should be tested to avoid healthy problems as to interfere with medical devices.
8. Technology neutrality: the functionalities should be open to any device with similar technology.

These legal principles define the deployment of the system and technology and devices to be used, they impose several requirements on software development and they bring a new way to define AAL applications.

Acknowledgements. This work has been partially supported by Projects CICYT TIN2008-06742-C02-02/TSI, CICYT TEC2008-06732-C02-02/TEC, SINPROB, CAM CONTEXTS S2009/TIC-1485 and DPS2008-07029-C02-02.

References

1. Schmidt, A.: Interactive context-aware systems interacting with ambient intelligence. IOS Press, Amsterdam (2005)
2. Emiliani, P., Stephanidis, C.: Universal access to ambient intelligence environments: Opportunities and challenges for people with disabilities. IBM Systems Journal 44(3), 605–619 (2005)
3. World population prospects: The 2006 revision and world urbanization prospects: The revision. Technical report, Population Division of the Department of Economic and Social Affairs of the United Nations Secretariat (last access: February 28, 2009)
4. Rentto, K., Korhonen, I., Vaatanen, A., Pekkarinen, L., Tuomisto, T., Cluitmans, L., Lappalainen, R.: Users' preferences for ubiquitous computing applications at home. In: First European Symposium on Ambient Intelligence 2003, Veldhoven, The Netherlands (2003)
5. Becker, M., Werkman, E., Anastasopoulos, M., Kleinberger, T.: Approaching ambient intelligent home care system. In: Pervasive Health Conference and Workshops 2006, pp. 1–10 (2006)
6. Floeck, M., Litz, L.: Integration of home automation technology into an assisted living concept. Assisted Living Systems-Models, Architectures and Engineering Approaches (2007)
7. Fraile, J., Bajo, J., Corchado, J.: Amade: Developing a multi-agent architecture for home care environments. In: 7th Ibero-American Workshop in Multi-Agent Systems (2008)
8. That right is enshrined in Article 12 of Universal Declaration of Human Rights, Article 7 the Charter of Fundamental Rights of the European Union (2000/C 364/01) and implicity in Fourth Amendment
9. See. European Court of Human Rights, López Ostra v. Spain - 16798/90 [1994] ECHR 46 (December 9, 1994). Katz v. United States, 389 U.S 347 (1967); Skinner v. Railway Labor Executives' Ass'n, 489 U.S. 602 (1989); To see differences between legal systems: Kirtley: Is implementing the EU Data Protection Directive in the United States irreconcilable with the First Amendment? Government Information Quaterly 16(2), 87–91 (2001)
10. Jhon, W.: Biometric scanning, law & policy: identifying the concerns-drafting the biometric blueprint. U. Pitt. L. Rev. 59, 97–155 (1997-1998)
11. Beck, U.: La sociedad del riesgo: hacia una nueva modernidad (1998)
12. Lin, C.H., Liou, D.Y., Wu, K.W.: Opportunities and challenges created by terrorism. Technological Forecasting and Social Change 74(2), 148–164, 158 (2007)
13. Kennedy, G.: Thumbs up for biometric authentication. Computer Law Review & Tech. 8, 379–407 (2003-2004)
14. Parejo Alfonso, Luciano: Seguridad pública y policía administrativa de seguridad, Valencia (2008)
15. To see examples, `http://www.biometrics.gov/Documents/FAQ.pdf` (040809)

16. That rights are enshrined in Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data OJ L 281, pp. 31–50 (23.11.1995) and Directive 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector OJ L 201, pp. 37–47 (31.7.2002) In the United States doesn't exist general regulation for data protection. Kuner, C.: An international legal framework for data protection: issues and prospects. Computer Law & Security Review 25, 307–317 (2009)
17. Haas, E.: Back to the future? The use of biometrics, its impact on airport security, and how this technology should be governed. Journal of Air Law and Commerce (69), 459 (Spring 2004)
18. Rodríguez de Santiago, J.Mª.: La ponderación de bienes e intereses en el Derecho Administrativo, Madrid (2000)
19. Kennedy. Note 20
20. Star, G.: Airport security technology: is the use of biometric identification technology valid under the Fourth Amendment? Law & Technology Journal 251 (2001-2002)
21. Luther, J.: Razonabilidad y dignidad humana. Revista de derecho constitucional europeo 7, 295–326 (2007)
22. Cilla, R., Patricio, M.A., Berlanga, A., García, J., Molina, J.M.: Non-supervised Discovering of User Activities in Visual Sensor Networks for Ambient Intelligence applications. Special session Challenges in Ubiquitous Personal Healthcare and Ambient Assisted Living, ISABEL (2009)
23. Sánchez-Pi, N., Molina, J.M.: A smart solution for elders in ambient assisted living. In: Mira, J., Ferrández, J.M., Álvarez, J.R., de la Paz, F., Toledo, F.J., et al. (eds.) IWINAC 2009, Part II, LNCS, vol. 5602, pp. 95–103. Springer, Heidelberg (2009)
24. Sánchez-Pi, N., Molina, J.M.: A centralized approach to an ambient assisted living application: An intelligent home. In: Omatu, S., Rocha, M.P., Bravo, J., Fernández, F., Corchado, E., Bustillo, A., Corchado, J.M. (eds.) IWANN 2009. LNCS, vol. 5518, pp. 706–709. Springer, Heidelberg (2009); Proceedings of International Workshop on Ambient Assisted Living (IWAAL 2009)

An Intelligent Tutoring System Oriented to the Integration of People with Intellectual Disabilities

A. Conde, K. López de Ipiña, M. Larrañaga, J.A. Elorriaga, J.M. López, E. Irigoyen, N. Garay-Vitoria, A. Ezeiza, and J. Rubio

Abstract. The development of Intelligent Tutoring Systems (ITS) based on mobile platforms offers new perspectives for a better integration of people with intellectual disabilities. The LAGUNTXO System aims to achieve the performance of human tutors. Due to the wide diversity related to a person with disabilities, an intelligent structure that may achieve a convenient tutoring system configuration for

---

A. Conde · K. López de Ipiña · E. Irigoyen · A. Ezeiza
Departamento Ingeniería de Sistemas y Automática, Escuela Técnica superior de Ingeniería, Bilbao, Escuela Politécnica Donostia, Grupo de Inteligencia Computacional
e-mail: {angel.conde,karmele.ipina,eloy.irigoyen, aitzol.ezeiza}@ehu.es

M. Larrañaga · J.A. Elorriaga
Departamento de Lenguajes y Sistemas Informáticos, Grupo GALAN, Facultad de Informática
e-mail: {mikel.larranaga,jon.elorriaga}@ehu.es

N. Garay-Vitoria
Departamento de Arquitectura y Tecnología de Computadores, Laboratorio de Interacción Persona Computador para Necesidades Especiales, Facultad de Informática
Universidad del País Vasco/Euskal Herriko Unibertsitatea
Plaza de Europa1, Donostia, 20008
e-mail: nestor.garay@ehu.es

J.M. López
Departmento de Informática e Ingeniería Industrial, Grupo de Investigación en Interacción Persona Ordenador y Bases de Datos (GRIHO)
Universitat de Lleida, Pl. Víctor Siurana 1, 25003, Lleida
e-mail: juanmi@diei.udl.cat

J. Rubio
Fundación LEIA CDT
Industrial Safety Unit, P.T. Alava – C/ Leonardo Da Vinci, 11,
01510 Miñano (Álava) - Spain
e-mail: jokinr@leia.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 639–647.
springerlink.com

each case has been incorporated. With an appropriate design of the structure and architecture of this task handler, it is very easy to operate by stakeholders. An automaton-based mechanism has been performed to technologically adapt the large amount of possibilities related to the interaction between people with disabilities, the task that is going to be made autonomously by users, and the mobile system elements. In this paper, LAGUNTXO architecture, operational ways, heuristic evaluation and a pilot study with final users are presented.

Keywords: People with intellectual and physical disabilities, Intelligent Tutoring System, Integration enhancement.

1 Introduction

Integrating people with disabilities into working and social environments is one of the main issues in applying Information and Communication Technologies (ICTs) into the Assistive Technology field. Particularly, it is necessary to pay special attention to the integration problem of people with intellectual disabilities. The origin of this project lies in a request of GUREAK ARABA S.L. (GRUPO GUREAK), a company that works in the integration of people with disabilities.

Computer aided systems have been successfully applied in many fields [1]. Intelligent Tutoring Systems (ITSs) are computer-based instructional systems with models of instructional content that specify what to teach, and teaching strategies that specify how to teach [2, 3]. The ITS monitors the learner performance to determine the student's mastery on certain topics or tasks and how to satisfy their requirements by selecting the most appropriate pedagogical strategy and content to be taught. Intelligent Tutoring Systems working into mobile platforms are an appropriate response to one of the main problems of people with disabilities: their integration into social and working environments [4]. These devices are designed to reach the user adaptation and to obtain an interaction that allows overcoming personal disabilities, for increasing the performance, individual autonomy, working capability, personal safety and a healthy environment in workplaces [5].

At the moment, the project has been carried out by a multidisciplinary re-search group with researchers from different fields such as Computing, Psychology, Medicine, and Engineering. These studies have also caused several works with social environment associations and companies devote to the industrial integration of people with disabilities. The level of success achieved within the project life is described in detail in the next sections. Section 2 describes in detail the architecture of LAGUNTXO SYSTEM. Section 3 shows results that have been obtained from the heuristic evaluation and a pilot study with final users. Finally, some conclusions and future work are shown in section 4.

2 Architecture of LAGUNTXO System

The LAGUNTXO system has been developed for the purpose of enhancing the integration of people with intellectual disabilities into their working and social environments. It has to achieve the performance of human tutors, going a further step than those classical tutoring systems [3] by dealing not only with the management of the tasks to be performed but also with the broad diversity of people with cognitive disabilities. Hence, the designed LAGUNTXO prototype has been structured in four main sub-systems: the Task Management System (TMS), the Intelligent Tutoring System in Mobile Platform (ITS-MP), the Intelligent Dialog System (IDS), and the Human Emotions Analysis System (HEAS). The Task Management System (TMS) has been developed to overcome the lack of suitable tools that deal properly with the broad diversity of people with disabilities when defining the tasks to be performed.

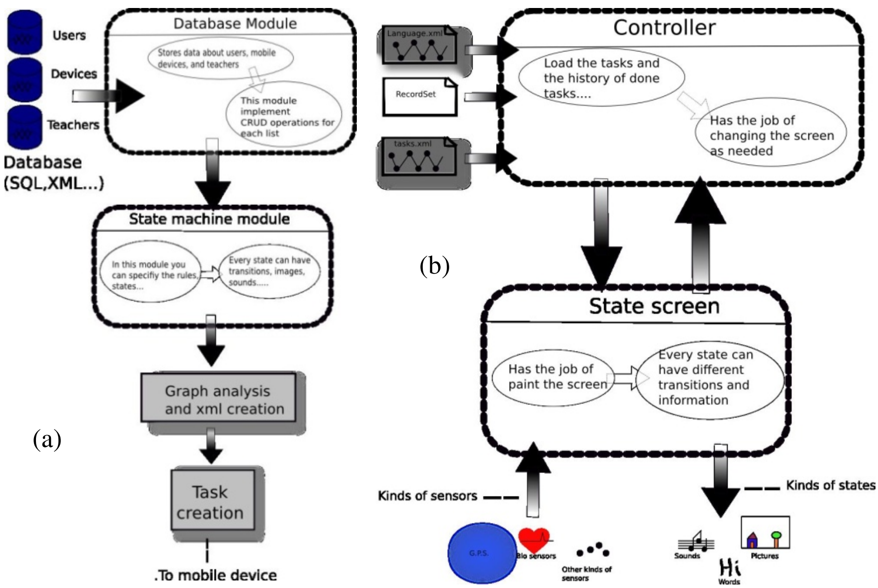


Fig. 1 (a) Structure of the TMS, (b) Structure of the ITS-MP

In this work, a system that provides the possibility of personalized configuration has been created: it handles any possible case in separate profiles due to the diversity of the disabled people collective.

The TMS is composed by three modules where the information is divided in the following parts: Tasks to carry out, Ergonomic characteristics and Users' personal information: to prevent accidents and emotional blockage situations for avoiding personal and physical damage.

In the TMS (Figure 1.a) this information is organized in several databases or repositories which can be continuously updated by users, tutors, caregivers and relatives. Moreover, the TMS has been designed to allow a comfortable and simple configuration, giving to users an easy way to build the profile that will be loaded into ITS-MP (Figure 1.b). In this sense, TMS has inside an automaton-based mechanism supporting several functions. First, the automaton handles the communication with tutors, caregivers and relatives to allow better understanding of its functionality. Finally, it generates the characteristics map of all Intelligent Tutoring configurable devices which will be activated. Finally, Figure 2a shows an edition screen to configure automaton states. These states are organized in several levels which are connected by conditional transitions. Each state represents a different subtask (simple or complex) to perform in order to solve the entire tasks. Depending on user disabilities, tasks profiles, and handheld mobile platforms, appropriate states and transitions will be charged on those platforms. Furthermore, the appropriate media type (image, sound, etc.) will be used considering both the user profile and the device features. TMS manages information about people with disabilities and the existing mobile platforms.

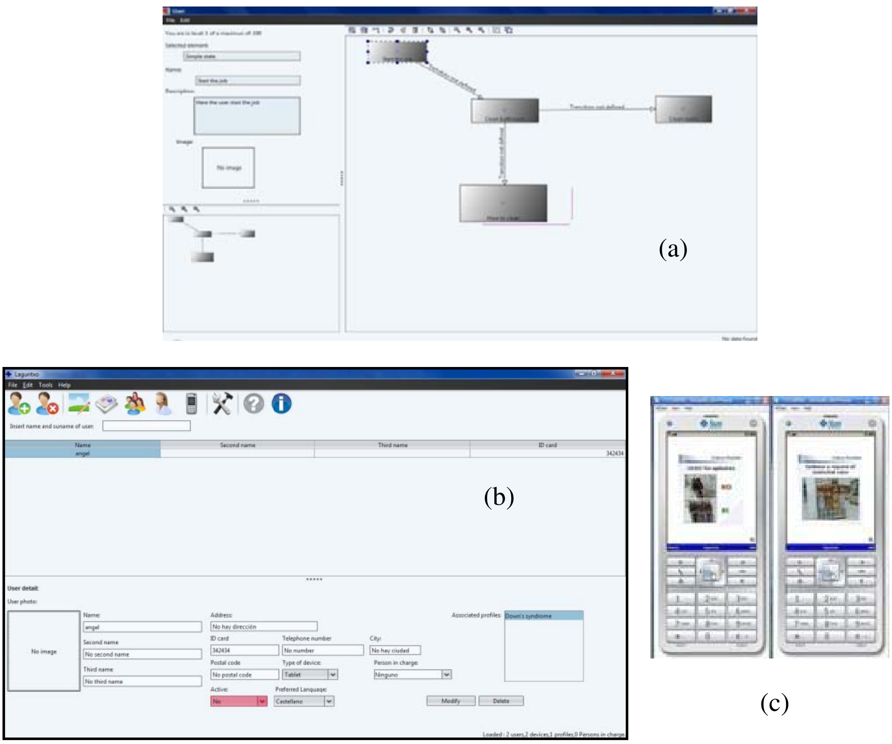


Fig. 2 **(a)** Interface to create the automata (b) Main Screen and Control Panel for configuring the TMS (c) ITS-MP interfaces

For every device, information that might be used to determine if it is appropriate for a particular user or characteristic is provided. Meanwhile, the users records contains their personal information in an anonymous way, as well as the personal tasks involved. This information is encrypted with different levels of access by password. Database structure is composed by several states. These states have items as images, videos, texts, etc., configuring the skeleton of the task. There exist two types of states. For interconnecting states, different transitions, defined by the stakeholders, have been created. The program is user-friendly, reliable, usefulness, agreeable, with a clear interface to be used by people with low computing knowledge. These interfaces are presented on tables or menus, depending on handled data (Figure 2).activated. Looking at ITS-MP in Figure 1b, and due to the characteristics of the people who will use these devices, it is absolutely necessary to design an interface that shows the following features: friendly, comfortable, flexible and ergonomically adapted to their characteristics. The main objective of this project lays in providing these users with a cognitive tool that contributes to the improvement of their autonomy, quality of life as well as help in the occupational health and safety management. Another objective is to integrate a task management into the portable device. To improve this, intelligent technologies based on fuzzy systems are used [2]. The basic structure of ITS-MP is based on rules and on an automaton mechanism to communicate with all sub-system of the ITS, similar to the mechanism previously detailed. An interface example can be seen at Figure 2c. It shows several screens where users may select the next subtask step by audio-video messages.

3 Evaluation

3.1 Heuristic Evaluation

Heuristic evaluation is a well-known and used usability evaluation method. To apply this technique, a group of experts must evaluate the fulfilment of a set of principles (or heuristics) by the user interface of the system to be evaluated [6]. In this sense, a Heuristic Evaluation has been carried out on the user interface of LAGUNTXO system. The evaluation process has been divided in two parts. First, the user interface of the tutor generator was evaluated using the set of heuristics proposed by Pierotti [7]. Second, the user interface of a concrete tutor was evaluated in a mobile phone, using the eight heuristics principles defined by Bertini [8]. A ninth heuristic was incorporated to evaluate learnability, as it is considered an important issue for LAGUNTXO system. Each heuristic principle was composed by a set of subheuristics that evaluators had to evaluate by following the Severity Ranking Scale (SRS) proposed by Nielsen [6], which ranks the potential usability errors according to their importance.

A total of 12 experts participated in the evaluation from different fields such as Computing, Psychology, Medicine, and Engineering. Every expert evaluated both ITS-TMS and ITS-MP systems, each of them using its corresponding set of heuristics and subheuristics. The procedure established that every expert had to pass all the heuristic and their corresponding subheuristics to the user interface. For every subheuristic, as a first step, experts had to check whether it was applicable or not to the user interface been studied. In case the subheuristic was applicable, the experts had to rank it according to Nielsen's SRS, ranging from 0 (no usability problem at all) to 4 (usability catastrophe). Not applicable subheuristics have not been represented in the results of the evaluation as the focus was set on identifying potential usability errors on applicable subheuristics.

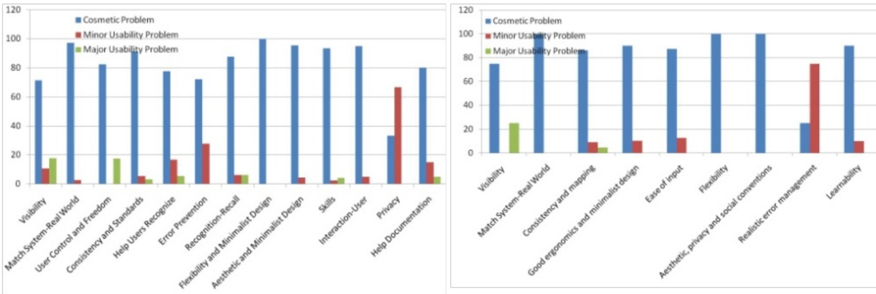


Fig. 3 Percentages of Cosmetic, Minor and Major usability problems found for ITS-TMS (a) and ITS-MP (b)

In order to normalize the results, values have been divided into three groups. Every group has been represented regarding its percentage for every subheuristic, so their relevance for every subheuristic can easily be seen in the figure. A first group was composed by subheuristics that are not considered as potential usability errors and the ones considered as cosmetic problems; this group has been represented in the figure with blue color. On the other hand, brown color represented the subheuristics identified as minor usability problems. Finally, major and catastrophic usability problem have been represented by using green color in the figure.

Figure 3 shows the results of the heuristic evaluation for both ITS-TMS and ITS-MP systems. Results show that studied systems have achieved a remarkable usability level. Even if there still are a number of major usability issues to solve, their amount is significantly low compared to the amount of subheuristics applicable on both studied user interfaces. Moreover, the sets of heuristics employed comprised 292 and 48 subheuristics respectively and have been based on solid foundations found in literature. This fact indicates that performed evaluation has been complete enough to ensure the validity of employed sets of heuristics and subheuristics. Besides, according to Nielsen [6], a number of five expert evaluators is enough to discover a relevant amount of usability problems in a user interface. In this sense, the amount of experts involved in the evaluation ensures that a relevant number of usability problems belonging to the user interface have been identified in the evaluation process.

3.2 Pilot Test

So as to test a first version of the ITS, we developed a prototype which has been used with people with intellectual disabilities and with people in risk of exclusion, both in working and collective environments. This prototype has been integrated into a mobile phone, as shown in Fig. 4. For evaluation purposes a module that collects data of the use of the tutor was developed. Initially, 14 users participated in the test: 7 people were chosen in indoor working places, and other 7 in a flat dedicated to the learning of daily life skills.

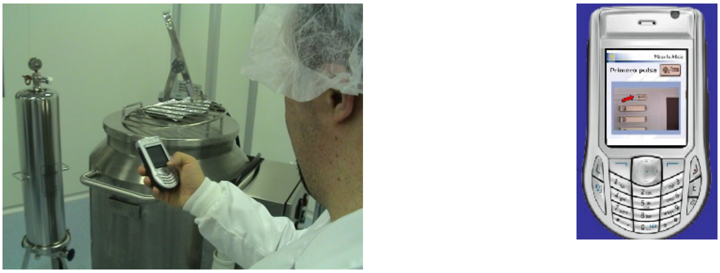


Fig. 4 An user of the ITS by a mobile phone

From the beginning of each test the implication of users and caregivers or working trainers was sought. Different indicators covering both subjective and objective parameters were created and measured during the test.

This allowed the project team to evaluate the usability of the tools taking into account the efficacy and efficiency of the tool and the satisfaction of the people implicated in the test. With relation to objective indicators (Table1), results show that tutor help was mainly used during first days and that performance is rapidly enhanced when using the tutor. Both caregivers/human tutors and researchers involved in this work expressed a remarkable satisfaction and considered it very useful. It is worth mentioning the short cycle of use and that it is used even outside the workinghours.

Table 1 Evaluation items and (b) subjective and (c) objective indicators

		Work activities	
		User	Initial/final interviews and questionnaires
Subjetive	Satisfaction	Trainer	Initial/final interviews and questionnaires
		Client	Final interviews
	Efficiency	Trainer	Number of calls
Objetive		and client	Number of questions
	Efficacy	Trainer	% os success in tasks

Subjective Indicators			
Attribute	User	Trainer	Client
Satisfaction	Very high	High	High
Usefulness	Very high	High	High
Improve the training	High	Moderate	
Improve the autonomy	High	Moderate	
Improve speed	Low	Low	
Reduce errors	high	Moderate	

Objetive Indicators	
Task performance	Most successfully finished
Questions	Few/first days
Phone Calls	Few first days

5 Concluding Remarks and Future Outlines

A Due to the wide casuistic of the people with intellectual disabilities a friendly, comfortable, flexible and ergonomically adapted Intelligent Tutoring System into mobile platforms has been designed. A prototype called LAGUNTXO has been developed (free software under GNU license). It works as an active distributed support system, and allows compensating user disabilities through task programming, facilitating suspension and resuming of tasks, offering help in blockage situations.

Heuristic evaluation and pilot tests indicate a high level of satisfaction of both the users and the stakeholders. On the one hand, it is increasing the users' autonomy, improving their training and the quality of their work. On the other hand, stakeholders have a new and easily configurable tool that reduces the time they have to dedicate in training.

Moreover, the job made so far has offered an opportunity to establish a solid link of communication between the social world of disabled people, caregivers and trainers, and the technological world of researchers. This link is leading to improve the knowledge and to overcome the gap between these two often separated worlds. In these sense, work is already being done to fulfil the new challenges that the project collaboration is generating. Complementary tools are being integrated in the system to improve its usefulness and to include the system within an ambient intelligence architecture. These tools are based on pattern matching (images and speech), human emotional feeling analysis, contextual information deployment (GPS, networking), and Artificial Intelligent (AI) techniques, giving the system the capacity of dynamic adaptation to the learning process.

Acknowledgements. The involved work has received financial support from the Spanish Ministerio de Ciencia y Tecnología, project TSI2006-14250-C02-01, and the Spanish Ministerio de Industria, Turismo y Comercio, project PDM-2006-178.

References

1. Sammour, G., et al.: The role of knowledge management and e-learning in professional development. International Journal of Knowledge and Learning 4(5), 465–477 (2008)
2. Wenger, E.: Artificial Intelligence and Tutoring Systems. Morgan Kaufmann, Los Altos (1987)
3. Yazdani, M.: Intelligent tutoring systems: an overview. In: Lawler, C.R., Yazdani, M. (eds.) Learning environments & tutoring systems, pp. 182–201. Ablex, Norwood (1987)
4. Conde, A., López de Ipiña, K., Larrañaga, M., Garay-Vitoria, N., Irigoyen, E., Ezeiza, A., Rubio, J.: LAGUNTXO: A rule-based intelligent tutoring system oriented to people with intellectual disabilities. In: Lytras, M.D., Damiani, E., Carroll, J.M., Tennyson, R.D., Avison, D., Naeve, A., Dale, A., Lefrere, P., Tan, F., Sipior, J., Vossen, G. (eds.) WSKS 2009. LNCS (LNAI), vol. 5736, pp. 186–195. Springer, Heidelberg (2009)
5. Kenny, C., Pahl, C.: Personalised correction, feedback and guidance in an automated tutoring system for skills training. International Journal of Knowledge and Learning 4(1), 75–92 (2008)

6. Nielsen, J.: Heuristic evaluation. In: Nielsen, J., Mack, R.L. (eds.) Usability Inspection Methods. John Wiley & Sons, New York (1994)
7. Pierotti, D.: Heuristic Evaluation - A System Checklist, http://www.stcsig.org/usability/topics/articles/he-checklist.html (Retrieved December 2009)
8. Bertini, E., Gabrielli, S., Stephen, K.: Appropriating and assessing heuristics for mobile computing. In: AVI 2006, pp. 119–126 (2006)

An Automatic Programming ACO-Based Algorithm for Classification Rule Mining

Juan Luis Olmo, José Mara Luna, José Raúl Romero, and Sebastián Ventura

Abstract. In this paper we present a novel algorithm, named GBAP, that jointly uses automatic programming with ant colony optimization for mining classification rules. GBAP is based on a context-free grammar that properly guides the search process of valid rules. Furthermore, its most important characteristics are also discussed, such as the use of two different heuristic measures for every transition rule, as well as the way it evaluates the mined rules. These features enhance the final rule compilation from the output classifier. Finally, the experiments over 17 diverse data sets prove that the accuracy values obtained by GBAP are pretty competitive and even better than those resulting from the top Ant-Miner algorithm.

1 Introduction

Data Mining (DM) entails the process of applying specific algorithms for extracting comprehensible, non-trivial and useful knowledge from data. The DM classification task aims to obtain a set of classification rules (a classifier) from a training data set. Once the classifier is built, one can apply these rules to other uncategorized data in order to label each instance with one of the predefined classes. The performance of the classifier is typically measured with the accuracy obtained when applying the classifier to a separate test set.

Support vector machines and neural networks have demonstrated to be accurate solutions to build classifiers. However, they have the disadvantage of generating non-linear classifiers. In constrast, logic based algorithms (i.e., decision trees and rule-based classifiers) provide more interpretability, but they are not as accurate as the previous methods [7].

Juan Luis Olmo · José Mara Luna · José Raúl Romero · Sebastián Ventura
Dept. of Computer Science and Numerical Analysis, University of Cordoba, Albert Einstein building, Rabanales Campus, 14071 Cordoba, Spain
e-mail: {juanluisolmo,i32luarj,jrromero,sventura}@uco.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 649–656.
springerlink.com

Ant Colony Optimization (ACO) [4] is a nature-inspired optimization metaheuristic based in the behavior and organization of ant colonies in their search for food. Ant algorithms have been successfully applied to a broad range of domains, including the extraction of classification rules in DM. For example, Ant-Miner, originally proposed by Parpinelli and colleagues [10], was the first algorithm based in ACO applied to the classification task. Ant-Miner follows a sequential-covering approach and has become a top algorithm in this field.

Furthermore, automatic programming is a method that uses search techniques to construct automatically a program that solves a given problem automatically, without requiring the user to know the structure of the solution. In fact, the problem is solved by simply specifying the goals to be reached. Typical examples of this method are Genetic Programming (GP) [2, 8] and Ant Programming (AP) [11], which uses ACO as search technique. The former has demonstrated that is capable to provide good performance for the design of classifiers, but the latter has never been used to tackle classification problems.

In this work we explore the application of an AP algorithm for classification rule mining. Thus, our proposal generates a rule-based classifier, which is composed of a set of classification rules that take the form *IF* <antecedent> *THEN* <consequent>. Our algorithm aims to construct accurate but also comprehensible classifiers, and first results show that it achieves good performance in terms of accuracy.

The remainder of the paper is organized as follows. In the next section we describe the proposed algorithm. Section 3 explains the experiments carried out and the data sets employed. In Section 4 we discuss the results obtained. Finally, some concluding remarks and ideas for future work are provided in Section 5.

2 The GBAP Algorithm

In this section we introduce Grammar Based Ant Programming (GBAP) algorithm.

Roughly speaking, GBAP follows a grammar guided automatic programming approach. This kind of systems are restricted by a defined grammar to ensure that any solution found is syntactically valid. The goal of GBAP is to obtain a classifier for a given data set, instead of a generic solution that could be applied to other data sets. This classifier takes the form of a decision list where discovered rules are sorted in descending order by fitness, and the bottom rule added to the classifier, which corresponds to the majority class in the data set, acts as default rule.

In ACO-based algorithms there must be an environment where ants cooperate each other. In GBAP, this environment is defined by the search space comprising all the possible expressions or programs that can be derived from the grammar. The space of states adopts the form of a derivation tree, and the path followed by the artificial ant could be seen as the sequence of derivation steps that leads the ant to a final state or solution.

In next sections we explain how classification rules are represented, presenting also a detailed pseudocode and description of the main characteristics of GBAP.

2.1 Rules Encoding

GBAP follows the *ant=rule* (i.e., *individual=rule*) approach [5]. Notice that once the ant is created, it just represents the antecedent of the new rule. In Section 2.2 we will analyze how the consequent is properly assigned to the rule.

GBAP prescribes a context-free grammar for the representation of the individuals, as shown in Figure 1. Notice that this grammar is expressed in prefix notation and should be always derived from the left. It implies that each transition from a state i to another state j is triggered after applying a production rule to the first non-terminal symbol of the state i. This design decision was taken because of performance reasons, in order to expedite the calculations neccessary to compute the rule fitness.

```
V = { <EXP>, <COND> }
Σ = { AND, =, !=, attr_1, attr_2, ..., attr_n,
          value_11, value_12, ..., value_1m,
          value_21, value_22, ..., value_2m,
      ..., value_n1, value_n2, ..., value_nm }
R = { <S> := <EXP>,
      <EXP> := AND <EXP> <COND> |
               <COND>
      <COND> := all possible valid combinations of the
                ternary operator-attribute-value }
S = { <S> }
```

Fig. 1 Context-free grammar used in GBAP, defined by $G = (V, \Sigma, R, S)$. Notice that any production rule consists of a left hand side (LHS) and a right hand side (RHS). The LHS always refers to a non-terminal symbol that might be replaced by the RHS of the rule (composed of a combination of terminal and non-terminal symbols).

2.2 Pseudocode and Main Characteristics of GBAP

An important characteristic of GBAP is the incremental generation of the space of states. In fact, depending on both the problem addressed and the number of derivations from the grammar permitted, it may be unfeasible to keep in memory the whole space of states. We therefore follow an incremental build approach in which as the ants are created we store the states they visit. This requires that each ant stores first the followed path. For this reason, the initial space of states is empty and all the possible transitions have the same amount of pheromones.

Another important characteristic of the algorithm proposed is that it considers two complementary heuristic measures. A first one is the *cardinality of the production rules (P_{card})*, and it is used due to the shape adopted by the space of states. This measure increases the probability of choosing transitions that lead to a greater number of solutions, and it is based on the cardinality measure proposed in [6]. When initializing the grammar in the algorithm, a cardinality table for the maximum number of derivations allowed is computed per each production rule. Given a state i and

Algorithm 1. High level pseudocode of GBAP

Require: *numberOfGenerations*, *numberOfAnts*
Initialize space of states, starting up the grammar
Create the classifier
for $i = 0$ **to** $i = numberOfGenerations$ **do**
  Create list $ants \leftarrow \{\}$
  for $j = 0$ **to** $j = numberOfAnts$ **do**
    $ant \leftarrow$ Create new ant
    Store *ant*'s path states in the space of states
    Evaluate *ant*, computing its fitness for each available class in the data set
    Add *ant* to the list *ants*
  end for
  Do niching algorithm, assigning the consequent to the ants and establishing the classifier rules
  for each *ant* in *ants* **do**
    if $fitness > threshold$ **then**
      Update pheromone rate in the path followed by *ant* proportionally to its fitness
    end if
  end for
  Evaporate the pheromone rate along the whole space of states
  Normalize values of pheromones
end for
Establish the default rule in the classifier
$predictiveAccuracy \leftarrow$ Compute the predictive accuracy obtained by the classifier when running over the test set
return *predictiveAccuracy*

all its possible subsequent states, the value of this heuristic for each possible transition is defined as the ratio between the number of solutions that can be successfully reached if the ant goes to the destination state applying this transition, and the number of all possible solutions that can be reached from the source state. Notice that this heuristic measure is only taken into account for intermediate transitions.

A second one is the *information gain* $(G(A_i))$. It is only used in transitions involving the application of production rules that imply the selection of attributes of the problem domain (i.e., $< COND > := operator - attribute - value$). This measure is similar to the one used by the Ant-Miner algorithm.

The use of both heuristic measures affects the creation process of new ants, when they move to the next state of their path. The transition rule will assign a probability to each available next state. In case of derivations that are not able to reach any final state in a number of steps less than or equal to the maximum number of derivations remaining, a probability equal to zero will be assigned and, in consequence, the ant will not select such a movement.

The probability that a given ant moves from a state i to another valid state j is defined by the following equation:

$$P_{ij} = \frac{\eta_{ij}^{\alpha} \cdot \tau_{ij}^{\beta}}{\Sigma_{i=0}^{j} \eta_{ij}^{\alpha} \cdot \tau_{ij}^{\beta}} \tag{1}$$

where α is the heuristic exponent, β is the pheromones exponent and η is computed as $G(A_i) + P_{card}$ (at least one of the two components must be equal to zero).

The fitness function that GBAP uses in the training stage for measuring the quality of the ants is the Laplace accuracy [3], which is defined as:

$$fitness = LaplaceAccuracy = \frac{1 + TP}{k + TP + FP} \tag{2}$$

where TP and FP stands for true positives and false positives, respectively, and k refers to the number of classes in the data set.

Concerning the assignment of the consequent, GBAP follows a niching approach analogous to that employed in [1], whose purpose is to evolve different multiple rules for predicting each class in the data set while preserving the diversity. Depending on each dataset and in the distribution of the instances by class, it is often not possible for a rule to cover all instances of a class and therefore it is necessary to discover additional rules for predicting this class. The niching algorithm takes care of it but it does not overlap with instances of another class. In addition, it is appropriate when removing redundant rules.

In the niching algorithm developed every instance in a data set is called a token, for which all ants in the colony will compete to capture. First of all GBAP computes an array of k fitness values per individual, one for each class (assuming that the respective class is assigned as consequent to the individual). Then, the following steps are repeated for each class: first, the ants are sorted by their respectively class fitness in descending order. Second, each ant tries to take as many tokens as it covers in case of tokens that belong to the computing class and also if the token has not been seized by other ant previously. Finally, the ant's adjusted fitness for this class is computed as:

$$adjustedFitness = fitness \cdot \frac{numberOfCapturedTokens}{numberOfClassTokens} \tag{3}$$

Once the k adjusted fitnesses have been calculated, the consequent assigned to each ant corresponds to the one that reports the best adjusted fitness. To conclude, individuals that have an adjusted fitness greater than zero –and consequently cover at least one instance of the train set– are added to the classifier.

Finally, regarding the pheromone update, if the quality of an ant is greater than a threshold value, then a delayed pheromone update over the path of this ant takes place. The threshold value has been fixed to 0.5 with the aim that bad solutions will never incluence the environment. The reinforcement is based on the quality of the solution encoded by the ant:

$$\tau_{ij}(t+1) = \tau_{ij}(1-\rho) + \tau_{ij} \cdot Q \cdot fitness \tag{4}$$

where τ represents the amount of pheromones in the transition from the state i to the state j; ρ, the evaporation rate; and Q is the parameter that permits to vary the influence of the reinforcement.

3 Data Sets and Preprocessing

This section describes the data sets used for the experimentation, as well as the preprocessing steps performed.

GBAP algorithm has been tested with many diverse data sets from the well-known UCI[1] machine learning repository. In fact, the data sets selected for the experiments present varied dimensionality, and some of them include missing values, while other do not, as shown in Table 1.

Table 1 Data sets description

DATASET	MISSING VALUES	INSTANCES	ATTRIBUTES			CLASSES
			Continuous	*Binary*	*Nominal*	
Hepatitis	yes	155	6	13	0	2
Sonar	no	208	60	0	0	2
Breast-c	yes	286	0	3	6	2
Heart-c	yes	303	6	3	4	2
Ionosphere	no	351	33	1	0	2
Horse-c	yes	368	7	2	13	2
Breast-w	yes	699	9	0	0	2
Diabetes	no	768	0	8	0	2
Credit-g	no	1000	6	3	11	2
Mushroom	yes	8124	0	0	22	2
Iris	no	150	4	0	0	3
Wine	no	178	13	0	0	3
Balance-scale	no	625	4	0	0	3
Lymphography	no	148	3	9	6	4
Glass	no	214	9	0	0	6
Zoo	no	101	1	15	0	7
Primary-tumor	yes	339	0	14	3	21

During the preprocessing stage we performed the following two actions using the Weka Machine Learning library[2]. Firstly, data sets comprising missing values were preprocessed replacing these values with the mode (in case of nominal attributes) and the arithmetic mean (in case of numerical attributes) of the entire data set [9]. Secondly, data sets with numerical attributes were properly discretized in order to only deal with categorical attributes [12].

Regarding the algorithm evaluation, we applied a stratified 10-fold cross-validation, so that the prediction performance is considered as the average accuracy over these 10 folds. In the stratified 10-fold cross-validation the entire data set is splitted into 10 mutually exclusive partitions, $P_1,...,P_k$, containing approximately the same number of patterns, and the same proportion of classes than the original data set.

[1] All data sets can be reached from the UCI website at `http://archive.ics.uci.edu/ml/datasets.html`

[2] The Weka library is publicly available at `http://www.cs.waikato.ac.nz/ml/index.html`

Table 2 Predictive accuracy(%) comparative results

Dataset	GBAP	Ant-Miner
Hepatitis	**82,17**	79,17
Sonar	**81,98**	74,70
Breast-c	71,40	**73,12**
Heart-c	**82,84**	76,62
Ionosphere	**93,02**	87,41
Horse-c	82,97	**84,38**
Breast-w	**96,50**	92,13
Diabetes	**75,80**	72,84
Credit-g	**70,79**	70,36
Mushroom	**98,26**	97,06
Iris	**96,67**	95,33
Wine	**97,01**	92,08
Balance-scale	**75,49**	68,09
Lymphography	**81,00**	77,69
Glass	**69,13**	64,43
Zoo	**95,60**	83,85
Primary	**37,91**	35,30

Then, 10 different experiments were executed using $\bigcup_{j \neq i} P_j$ as the training set at the ith-experiment and P_i as the test set.

4 Results

Experiments compare the performance of GBAP against Ant-Miner, in terms of predictive accuracy, over the data sets listed in Table 1. More specifically, Ant-Miner was used in all executions –10 per dataset– with its default parameters. For the GBAP algorithm its configuration parameters were set to: *number of ants* = 20, *number of generations* = 100, *max number of derivations* = 15, *initial pheromone amount* = 1.0, *evaporation rate* = 0.05, *min pheromone amount* = 0.1, *Q* = 1.0, *alpha* = 0.4, and *betha* = 1.0.

Table 2 summarizes the results obtained, where each row shows the data set tested and the resulting average accuracy (in %) for both algorithms. Best results per algorithm and dataset are highlighted in bold typeface.

As can be seen, GBAP obtains best results in approximately 88% of cases. We also analyzed the statistic significance of the obtained results applying the non-parametric Wilcoxon pair-test, and the results ($z = 3.195$, $p < 0.001$) proved that there are significant differences with a probability of 99%, where GBAP is significantly more accurate than Ant-Miner.

5 Conclusions and Future Work

In this paper we presented a novel automatic programming algorithm based in ACO restricted by the use of a context-free grammar for mining classification rules from diverse data sets. The proposal is supported by a two-sided heuristic function that guides the search process of the valid solutions, as well as the chance of modifying the complexity of rules mined by simply varying the number of derivations allowed for the grammar.

In this work we have compared the performance of GBAP against Ant-Miner in 17 different data sets publicly available. The obtained results prove that the former is significantly more accurate than the latter. As future work we plan to apply other problem-dependent heuristic measures. We also will explore the consideration of adding new functionality to deal with continuous attributes and adapting the current algorithm to the multi-objective approach.

Acknowledgments. This work has been supported by the Regional Government of Andalucia and Ministry of Science and Technology, projects TIC-3720 and TIN2008-06681-C06-03.

References

1. Ávila, J., Gibaja, E., Zafra, A., Ventura, S.: A niching algorithm to learn discriminant functions with multi-label patterns. In: Corchado, E., Yin, H. (eds.) IDEAL 2009. LNCS, vol. 5788, pp. 570–577. Springer, Heidelberg (2009)
2. Banzhaf, W., Nordin, P., Keller, R.E., Francone, F.D.: Genetic Programming - An Introduction; On the Automatic Evolution of Computer Programs and its Applications. Morgan Kaufmann, San Francisco (1998)
3. Clark, P., Boswell, R.: Rule induction with CN2: Some recent improvements. In: Kodratoff, Y. (ed.) EWSL 1991. LNCS, vol. 482, pp. 151–163. Springer, Heidelberg (1991)
4. Dorigo, M., Blum, C.: Ant colony optimization theory: a survey. Theoretical Computer Science 344, 243–278 (2005)
5. Espejo, P.G., Ventura, S., Herrera, F.: A survey on the application of genetic programming to classification. IEEE Transactions on System, Man and Cybernetics Part C (2008) (in press)
6. Geyer-Schulz, A.: Fuzzy Rule-Based Expert Systems and Genetic Machine Learning. Studies in Fuzziness, vol. 3. Physica-Verlag, Heidelberg (1995)
7. Kotsiantis, S.B., Zaharakis, I.D., Pintelas, P.E.: Machine learning: a review of classification and combining techniques. Artificial Intelligence Reviews 26, 159–190 (2006)
8. Koza, J.R.: Genetic programming: on the programming of computers by means of natural selection. The MIT Press, Cambridge (1992)
9. Larose, D.T.: Discovering Knowledge in Data: An Introduction to Data Mining. Wiley, Chichester (2005)
10. Parpinelli, R., Freitas, A.A., Lopes, H.S.: Data mining with an ant colony optimization algorithm. IEEE Trans. on Evolutionary Computation 6, 321–332 (2002)
11. Roux, O., Fonlupt, C.: Ant programming: or how to ants for automatic programming. In: Dorigo, M., et al. (eds.) ANTS 2000, pp. 121–129 (2000)
12. Witten, I.H., Frank, E.: Data Mining Practical Machine Learning Tools and Techniques. Morgan Kauffman, San Francisco (2005)

Energy Cost Reduction in the Synchronization of a Pair of Nonidentical Coupled Hindmarsh-Rose Neurons

A. Moujahid, A. D'Anjou, F. Torrealdea, and F.J. Torrealdea

Abstract. Many biological processes involve synchronization between nonequivalent systems, i.e, systems where the difference is limited to a rather small parameter mismatch. The maintenance of the synchronized regime in this cases is energetically costly [1]. This work studies the energy implications of synchronization phenomena in a pair of structurally flexible coupled neurons that interact through electrical coupling. We show that the forced synchronization between two nonidentical neurons creates appropriate conditions for an efficient actuation of adaptive laws able to make the neurons structurally approach their behaviours in order to decrease the flow of energy required to maintain the synchronization regime.

1 Introduction

When a given oscillator moves freely on its natural attractor its oscillatory regime consists of a balanced exchange of energy between the system and its environment that occurs spontaneously through the divergent components of the systems structure without concurrence of any additional device. If, on the other hand, the system

A. Moujahid
Computational Intelligence Group, Department of Computer Science, University of the Basque Country, Manuel de Lardizabal 1, 20018 San Sebastin, Spain
e-mail: jibmomoa@gmail.com

A. D'Anjou
Computational Intelligence Group, Department of Computer Science, University of the Basque Country, Manuel de Lardizabal 1, 20018 San Sebastin, Spain

F. Torrealdea
Master Program Tissue Engineering, University College, London WC1E6BT

F.J. Torrealdea
Computational Intelligence Group, Department of Computer Science, University of the Basque Country, Manuel de Lardizabal 1, 20018 San Sebastin, Spain

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 657–664.
springerlink.com

is forced to synchronize to a different guiding system its oscillatory regime occurs on an unnatural region of the state space where there is a nonzero net average exchange of energy with its environment. This net flow of energy per unit time requires the concurrence of a coupling device that includes an external source of energy. This flow of energy is necessary to maintain the synchronized regime and constitutes a cost for the synchronization process [1]. This consumption of energy can be reduced if the guided system itself adapts its structure to become closer to the one of the guiding system [3]. Ideally, if the systems become identical their joint dynamics is attracted toward a regime of zero error in the variables.

Many biological processes involve synchronization between different members of the same family of systems that have similar, although not identical, values of some distinctive parameters. This work studies the energy implications of synchronization phenomena in a pair of structurally flexible coupled neurons that interact through electrical coupling. We show that the forced synchronization between two nonidentical neurons creates appropriate conditions for an efficient actuation of adaptive laws able to make the neurons structurally approach each other in order to decrease the flow of energy required to maintain the synchronization regime. The neuron has been modelled by a four-dimensional Hindmarsh-Rose model [6, 7, 8, 9, 10]. This model is described by the following equations of movement:

$$\begin{aligned}
\dot{x} &= ay + bx^2 - cx^3 - dz + \xi I, \\
\dot{y} &= e - fx^2 - y - gw, \\
\dot{z} &= m(-z + s(x+h)), \\
\dot{w} &= n(-kw + r(y+1)),
\end{aligned} \tag{1}$$

where $a, b, c, d, \xi, I, e, f, g, m, s, h, n, k, r$, and l are the parameters that govern the dynamics of the neural system. The variable x is a voltage associated to the membrane

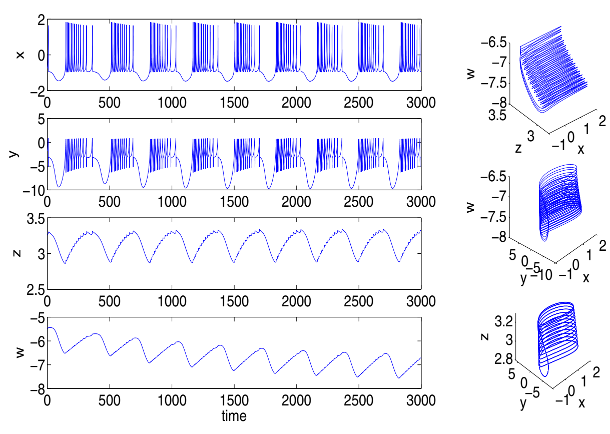


Fig. 1 Time series and 3D projections of the dynamical variables $x(t)$, $y(t)$, $z(t)$, $w(t)$ of the four-dimensional Hindmarsh-Rose neuron model

potential, variable y although in principle associated to a recovery current of fast ions has been transformed into a voltage, variable z is a slow adaptation current associated to slow ions, and variable w represents an even slower process than variable z [11]. I is a external current input, and is the main parameter we used to control the modes of spiking and bursting activity of the model. For the numerical results of this work we fix the parameters to the values $a = 1$, $b = 3.0(mV)^{-1}$, $c = 1(mV)^{-2}$, $d = 0.99M\Omega$, $\xi = 1M\Omega$, $e = 1.01mV$, $f = 5.0128(mV)^{-1}$, $g = 0.0278M\Omega$, $m = 0.00215$, $s = 3.966\mu S$, $h = 1.605mV$, $n = 0.0009$, $k = 0.9573$, $r = 3.0\mu S$, $l = 1.619mV$.

Figure 1 shows a chaotic time series of the four variables. The complexity achieved by the incorporation of a slow variable w that increases the realism of the description of slow Calcium currents can be observed in the projections of the attractor on the (x,y,z), (x,y,w) and (x,z,w) axes.

In Sec. 2 we report the energy-like function associated to a four-dimensional Hindmarsh-Rose model. Sec. 3 briefly summarizes the adaptation mechanism we used to adapt the structure of the postsynaptic neuron, and presents computational results of the synchronization process of two electrically nonidentical coupled neurons. We consider that the presynaptic (sending) neuron always signal in a chaotic regime, while the postsynaptic (receiving) neuron is set to its quiescent state at a low value of its external current. In a first stage, the postsynaptic neuron has been forced to synchronize with the presynaptic one, then we initiate an adaptive process that adapts some parameters of the postsynaptic neuron to ones of the presynaptic neuron. We have analysed the energy dissipation of the receiving neuron during the synchronization process without and with structural adaptation.

2 Four-Dimentional Hindmarsh-Rose Model Energy

In the Hindmarsh-model given by Eq. (1) the energy function $H(x)$ is given by [2]

$$H = \frac{p}{a}(\frac{2}{3}fx^3 + \frac{msd-gnr}{a}x^2 + ay^2) + \frac{p}{a}(\frac{d}{ams}(msd-gnr)z^2 - 2dyz + 2gxw) \quad (2)$$

where p is a parameter. As in the model time is dimensionless and every adding term in Eq.(2) has dimensions of square voltage, function H is dimensionally consistent with a physical energy as long as parameter p has dimensions of conductance. In this paper we fix parameter p to the arbitrary value $p = -1S$. The minus sign has been chosen to make consistent the outcome of the model with the usual assumption of a demand of energy associated with the repolarization period of the membrane potential and also with its refractory period (see Fig. 2).

And the corresponding energy derivative $\dot{H}$ is given by [2]

$$\dot{H} = \frac{2p}{a}\begin{pmatrix} fx^2 + \frac{msd-gnr}{a}x + gw \\ ay - dz \\ \frac{d}{ams}(msd-gnr)z - dy \\ gx \end{pmatrix} \begin{pmatrix} bx^2 - cx^3 + \xi I \\ e - y \\ msh - mz \\ nrl - nkw \end{pmatrix} \quad (3)$$

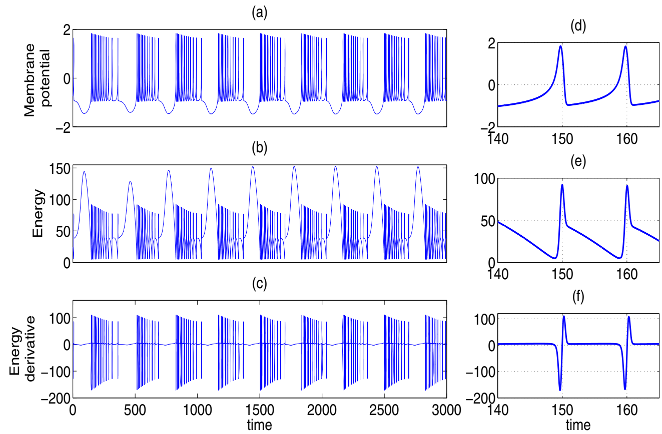


Fig. 2 (a) Action potentials, (b) energy and (c) energy derivative for the Hindmarsh-Rose model neuron. (d), (e) and (f) Details of the action potential, energy and energy derivative associated to two spikes.

is also dimensionally consistent with a dissipation of energy. As the states of an isolated HindmarshRose neuron are confined to an attractive manifold the range of possible values of its energy is recurrent and the long term average of its energy derivative is zero.

This energy and energy derivative functions are used to evaluate the energy consumption of the neuron in isolation and also when it is connected to other neurons through electrical synapses, and provide the basis for all the computational results presented in this work. The procedure followed to find this energy function has been reported in detail in [1].

Figure 2 (a) shows a series of action potentials (variable x in the model neuron). Fig. 2(b) and Fig. 2(c) show both energy and energy derivative corresponding to that action potentials. Fig. 2(e) and Fig. 2(f) show detail of energy and energy derivative associated to a train of two action potentials. For each action potential it can be appreciated (see Fig. 2(e,f)) that the energy derivative is first negative, dissipation of energy while the membrane potential depolarizes during the rising period of the spike, and then positive, contribution of energy to repolarize the membrane potential during its descending period. During the refractory period between the two spikes the energy derivative remains slightly positive, still demanding energy, until the onset of the following action potential.

3 Synchronization Energy of Two Electrically Coupled Neurons

In this section we analyze the energy aspects of the synchronization of two non-identical neurons coupled by an electrical synapse. The presynaptic neuron is set in the chaotic spiking-bursting regime corresponding to an external current $I_1 = 3.024$.

While the postsynaptic neuron is set to its quiescent state at a low value $I_2 = 0.85$ of its external current. The two neurons are coupled unidirectionally according to the following equations:

$$\begin{aligned} \dot{x}_i &= ay_i + bx_i^2 - cx_i^3 - dz_i + \xi I_i + K_i(x_j - x_i), \\ \dot{y}_i &= e - fx_i^2 - y_i - gw_i, \\ \dot{z}_i &= m(-z_i + s(x_i + h)), \\ \dot{w}_i &= n(-kw_i + r(y_i + 1)), \end{aligned} \tag{4}$$

where $K_1 = 0$ and $K_2 \geq 0$ is the coupling strength. $i, j = 1, 2; i \neq j$ are the indices for the neurons. Note that the coupling affects only the first variables x_2 of the postsynaptic neuron.

The coupling scheme given in Eq. 4 can be written in general terms as follows:

$$\begin{aligned} \dot{y} &= f(y, p) \\ \dot{x}_k &= f(x_k, q) + K(y - x_k), \end{aligned} \tag{5}$$

where p and q stand for the parameters of the presynaptic and postsynaptic neurons, and $x_k(t)$ indicates the states of the postsynaptic neuron when the coupling strength is set to k. Notice that $K(y - x_k)$ is the coupling interface required in order to be physically able to implement the coupling of both neurons $\dot{x} = f(x)$ and $\dot{y} = f(y)$.

If the coupling strength K is large enough as to make the errors in the variables $e = x_k - y$ small, an operational law that adapts the parameters of the postsynaptic neuron to the ones of the presynaptic neuron is given by [3]

$$\dot{e}_i^p = -\left[\sum_{l=1}^{n} \left(\frac{\partial f_l(x_k, q)}{\partial q_i} \right)_{(y,p)} e_l \right] \tag{6}$$

where $e^p = q - p$ denotes the vector of parameter errors, and the summation is over every component of the vector field f. The above law is general and can be used to find specific adaptive laws to any kind of homochaotic systems provided they are coupled through a feedback scheme of large enough coupling strength.

In the following we analyse the change in the balance of energy of the postsynaptic neuron when its external current parameter is governed by an adaptive law in order to reach the nominal value of the external current in the presynaptic neuron. The adaptation law has been implemented following Eq. 6. For this experiment, we have used a coupling strength with value $k = 5$. We started the adaptation procedure at $t = 100$ and registered data between $t = 50$ and 200 for proper observation of the evolution of both energy and energy derivative during the process. The registrated values has been averaged over a convenient length of time in order to avoid large fluctuations. The dissipated energy (energy derivative) has been averaged over five units of time, while the proper energy has been averaged over ten units of time.

Figure 3 shows the average values of both energy and energy derivative per unit time. In Fig. 3 (a) we can see that in a first stage ($t < 100$), the receiving neuron is forced to synchronize with the sending neuron, and oscillates in an unnatural region of the state space characterized by an average energy of about 66 (arb. units), and a nonzero energy derivative average of about 14 (arb. units), ie, a nonzero net average

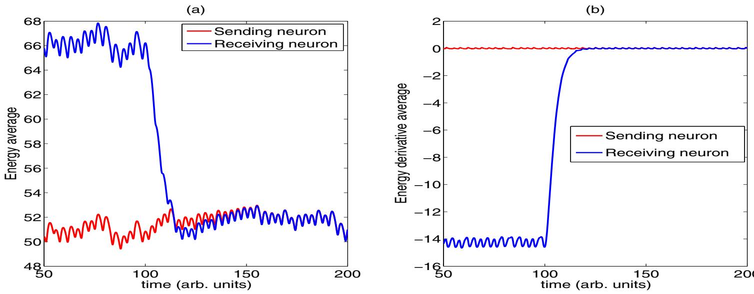


Fig. 3 (a) Average over ten units of time of the energy per unit time. (b) Average over five units of time of the energy derivative per unit time. Adaptation of the external current I_2 begin at $t = 100$. The coupling strength is set to value $k = 5$.

exchange of energy with its environment (see Fig 3 (b)). After adaptation take place, the two neurons become structurally close each other, and enter in a completely synchronized regime of balanced exchange of energy between the system and its environment corresponding to zero value of energy derivative ($\dot{H} = 0$).

To illustrate the ability of this adaptive laws to decrease the energy dissipation, we have computed, for the receiving neuron, the average energy and average energy dissipation over a convenient length of time at different values of the coupling strength ranging from $K = 0$ to $K = 2$. The adaptation run at time $t = 100$.

Figure 4 shows average over ten units of time of the energy per unit time at different values of the coupling strength ranging from $k = 0$ to $k = 2$. It can be seen that before the adaptation occurs ($t < 100$), and for low value of the coupling strength not sufficient to induce a certain degree of synchrony, the receiving neuron

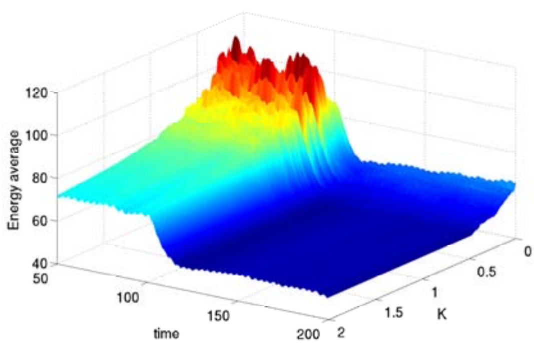


Fig. 4 Average over five units of time of energy per unit time of the receiving neuron at different values of the coupling strength K. Adaptation of the external current I_2 begin at $t = 100$.

shows a waving average energy pattern. This oscillating regime of energy average disappears once the adaptation process start, and decreases to low values. For high values of the coupling strength, the receiving neuron is forced to synchronize with the sending neuron and moves in a region of state space where the average of its energy is greater than that of the sending neuron. When the coupling strength is large enough as to make the errors in the variables small, and as soon as the adaptation process starts the average energy quickly decreases to values corresponding to the average energy of the sending neuron.

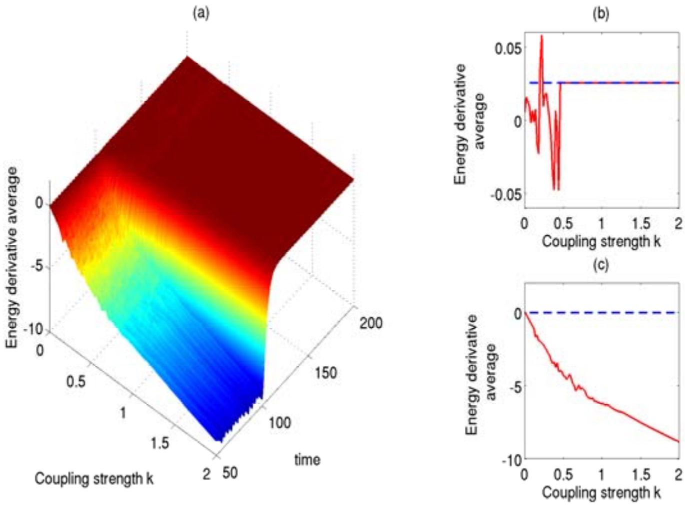


Fig. 5 (a) Average over ten units of time of of energy derivative per unit time of the receiving neuron at different values of the coupling strength K. (b) and (c) show respectively the average of energy derivative of the receiving neuron (solid line) and the sending neuron (dash line) as a function of the coupling strength with and without structural adaptation. Adaptation of the external current I_2 begin at $t = 100$.

Figure 5 shows average over five units of time of the energy derivative per unit time at different values of the coupling strength ranging from $k = 0$ to $k = 2$. When $K = 0$, ie, no guidance at all, the receiving neuron moves on its natural region of state space and its averaged dissipated energy is zero. As soon as the coupling device is connected the average energy derivative per unit time becomes negative, that is, it start to dissipate on average an energy that the coupling device will have to provide in order to maintain the forced regime. The required energy increases with coupling strength as it can appreciated in Fig.5 (c). Once the adaptation process starts the average energy derivative quickly decreases to zero, reflecting the fact the receiving neuron has become structurally so close to the sending neuron that they can reach a regime of identical synchronization.

4 Conclusion

This paper analyze the energy aspects of the forced synchronization between two noidentical electrically coupled neurons. We have considered that the presynaptic neuron always signal in a chaotic regime, while the postsynaptic neuron is set to its quiescent state. Under these conditions to keep the synchronized regime requires a net flow of energy that can be costly to maintain. We have shown that this energy cost can be reduced notably if the postsynaptic neuron is able to adapt its structure in order to approach the dynamics of the presynaptic neuron. Biological structures are particularly flexible in adapting their parameters and the mechanism of adaptation introduced in this work could make some the required collective behaviors energetically less costly.

References

1. Sarasola, C., Torrealdea, F.J., d'Anjou, A., Moujahid, A., Graña, M.: Energy balance in feedback synchronization of chaotic systems. Phy. Rev. E 69, 011606 (2004)
2. Torrealdea, F.J., Sarasola, C., d'Anjou, A., Moujahid, A., Vlez de Mendizbal, N.: Energy efficiency of information transmission by electrically coupled neurons. BioSystems 97, 60–71 (2009)
3. Sarasola, C., Torrealdea, F.J., d'Anjou, A., Moujahid, A., Graña, M.: Feedback synchronization of chaotic systems. Int. J. Bifurcation Chaos Appl. Sci. Eng. 13, 177–191 (2003)
4. Kepler, T.B., Marder, E., Abbott, L.F.: The effect of electrical coupling on the frequency of model neuronal oscillators. Science 248, 83 (1990)
5. Connors, B.W., Long, M.A.: Electrical synapses in the mammalian brain. Annu. Rev. Neurosci. 27, 393 (2004)
6. Hindmarsh, J., Rose, R.: A model of neural bursting using three coupled first order differential equations. Proc. R. Soc. Lond., Ser. B221, 87 (1984)
7. Rose, R., Hindmarsh, J.: A model of thalamic neuron. Proc. R. Soc. Lond., Ser. B225, 161 (1985)
8. Rose, R., Hindmarsh, J.: The assembly of ionic currents in a thalamic neuron. I. the three dimensional model. Proc. R. Soc. Lond., Ser. B 237, 267–288 (1989a)
9. Rose, R., Hindmarsh, J.: The assembly of ionic currents in a thalamic neuron. II. the stability and state diagrams. Proc. R. Soc. Lond., Ser. B 237, 289–312 (1989b)
10. Rose, R., Hindmarsh, J.: The assembly of ionic currents in a thalamic neuron. III. the seven dimensional model. Proc. R. Soc. Lond., Ser. B 237, 313–334 (1989)
11. Pinto, R.D., Varona, P., Volkovskii, A., Szcs, A., Abarbanel, H.D., Rabinovich, M.: Synchronous behavior of two coupled electronic neurons. Phys. Rev. E 62, 2644 (2000)

A Color Transformation for Robust Detection of Color Landmarks in Robotic Contexts

Ramón Moreno, Manuel Graña, and Alicia d'Anjou

Abstract. We present in this work a robust color transformation which has been applied succesfully to natural scenes allowing the fast and precise segmentation of regions corresponding to color landmarks under uncontrolled lightning. The process is grounded in the the Dichromatic Reflexion Model (DRM) and the properties of the RGB space.

Keywords: Reflection, Color Transformation.

1 Introduction

Robust and fast detection of color regions is one of the typical artificial vision problems. Given our color perception, color clustering is not an appropriate approach most of the times. Among the various color spaces, the HSV and CIE L*a*b [1] are the ones closest to human perception. We work on the RGB space, where the DRM model is defined [2].

The need to detect color regions steems from its conventional use in signaling: red for danger, blue and green for informative, yelow for danger advice. Also Red, Green and Blue are the basic colors in the RGB space unit cube. All the remaining colors are represented as linear combinations of these colors.

In robotic contexts, working on artificial environments, we must benefit from this information source by the robust detection of signaling symbols drawn in the basic colors. A critical problem is removing the reflections in the image, which interfere with the observed surface. The two goals of the color image processes are identified as: efficient color detection and reflection removal. We present an efficient solution to these problems on the RGB space, using the idea of Specular Free images [3,6].

In the following, section 2 presents a brief explanation of the DRM model and its justification in the RGB space. We present our method in section 3. We present some experimental results in section 4. We give our conclusions and further work lines in section 5.

Ramón Moreno · Manuel Graña · Alicia d'Anjou
University of the Basque Country UPV/EHU, Spain
Grupo de Inteligencia Computacional
http://www.ehu.es/ccwintco

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 665–672.
springerlink.com

2 Some Properties of the RGB Cube and the DRM

Human chromatic perception is the resul of biological evolution along millions of years. The mental interpretation of colors is subject to subjective aspects: philosophical, cultural and evolution. We can say that the human beings have developed individual color perception traits. However we have a consensus on the basic color interpretation which is represented in the color space used for their representation. The HSV color space is one that matches the human perception better than the RGB space. The pair Hue-Saturation defines the chromatic space, while V is the light intensity.

2.1 Some Properties of RGB

The most used color space is RGB. From the computational point of view, and the artficial vision one, the RGB space has the following interesting properties:

1. It is the default color representation space for all the machines, from perception (Bayer's mosaic) up to the monitor visualization.
2. The vertices of the unit RGB cube represent the primary colors (red, blued, green), the secondary colors (yellow, cyan, magenta) and the black and white colors. The ones most used in signalization.
3. The reflections or brights are characterized in the RGB cube for its proximity to the black-white diagonal.
4. The Dichromatic Reflection Model (DRM) has been defined in the RGB space.

2.2 Dichromatic Reflection Model (DRM)

The DRM was introduced by Shafer [2], explaining each point of the observed surface as the sum of two components: a diffuse component D and an specular component S, as can be appreciated in figure [1]. The diffuse component refers to the chromatic properties of the observed surface, while the specular component refers to the light color. Reflections in the surface have a great content of the specular component.

Algebraically, DRM is expressed as $I(x) = m_d(x)D + m_s(x)S$ where m_d and m_s are weighting values for the diffuse and specular components, respectively, and their values are in the range between 0 and 1. Therefore, a surface with homogeneous chromatic features can be expressed as the sum of two colors, its own and the illumination color. In figure 1 the shadowed region represents the convex region of the plane $\sqcap_{dc}$containing all the possible colors expressed by the above equation. For an scene with several colors, it will be expressed as $I(x) = m_d(x)D(x + m_s(x)S$, where D depends on the spatial localization x. However, still S is independent of the spatial localization x, assuming that the illumination chromaticity is constant for all the scene. Finally, the complete model is expressed as $I(x) = m_d(x)D(x) + m_s(x)S(x)$ where both chromaticities depend on the spatial localization: several surface and illumination colors.

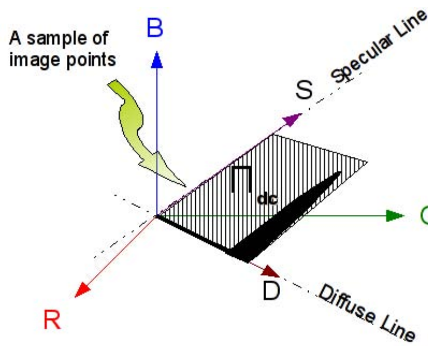


Fig. 1 Dichromatic Reflection Model

2.3 RGB Features from the DRM Point of View

The DRM expresses the color of each pixel as the sum of a difuse and specular components. Most authors assume a uniform illumination as a problem simplification. This simplification is correct, because most of the times the illumination color is close to the pure white. However, working in constant illumination chromaticity, we can obtain it used several methods [4,5].

We can classify image pixels into:

- Diffuse pixels: showing the observed surface color, with an almost null specular component.
- Specular pixels: whose specular component is much bigger than the diffuse component.

Placement of difuse and specular pixels is qualitative different in the RGB cube. Let us focus on the proximity of pixels to the black-white cube diagonal, defined as $L_w: (r, g, b) = P + s\vec{u}; \forall s \in \mathbb{R}$ where $P = [0,0,0]$ and $\vec{u} = [1,1,1]$. Given a uniform color region, without any specular component, its representation in the RGB cube would be a line, the diffuse chromaticity line for this region. However, due to noise, it appears as an elongated point cloud.

Given a uniform color region, with high specular component, from the DRM point of view, it must appear as a line parallel to line L_w or approaching it. Again, due to noise, an elongated pint cloud appears. Specular image regions have RGB representations far from the color space origin.

Finally, a uniform color region (color constancy) with some non negligible specular component must show a V shape. The point cloud beginning in the coordinate origin and go away from line L_w contain the diffuse points, while the ones close to it are the specular ones. Using this knowledge, we can penalize the specular component and magnify the diffuse component.

3 Method

Being interested in pure color regions, we expect their color representation in the RGB cube far from the line L_w . On the other hand, we want to penalize specular regions, those close to line L_w and far from the coordinate system origin.

A main feature of line L_w is that the three components of its points are equal $r = g = b; \forall r, g, b \in [0,1]$. For pixels close to this region, we $r \cong g \cong b; \forall r, g, b \in [0,1]$. As the pixels fall away from this line, the differences among their components are greater. We use this difference as the intensity of the processed image. As we want to preserve the chromatic information, only the intensity is modified, boosting the diffuse pixels and nullifying the specular pixels. The new intensity of the pixels is computed as difference between the maximum and minimum of their RGB components:

$$Intensity = max_{\{r,g,b\}}(I) - min_{\{r,g,b\}}(I)$$

This intensity replaces the V component in the HSV representation, thus preserving the chromatic content of the pixel. We show in algorithm 1 an implementation in SciLab.

Algorithm 1 SF2

```
//I is a RGB image
// IR is the transformed image

Function IR = SF2(I)
        New_Intensity = (max(I,3) – min(I,3));
        Imghsv = rgb2hsv(I);
        Imghsv(:,:,3) = New_Intensity;
        IR = rgb2hsv(Imghsv);
Endfunction
```

3.1 Application

The SF2 image, the one obtained after the described transformation, is characterized by the absence of reflections, substituted by dark spots. Also the difuse regions are boosted in the image. With an straightforward analysis we can find all the diffuse regions.

4 Experiments

We have performed experiments in three different contexts: first the detection of markers in real scenes, other with synthetic images, and the last about the detection of robots in real time. All the results can be viewed in the following web address: http://www.ehu.es/ccwintco/index.php/SMC

4.1 Mark Detection

The definition of the experiment is as follows:

1. Context:
   a) Place: a lab corridor, with artificial illumination of diverse intensity and uniform color.
   b) Markers are DIN A4 sheets of different colors: red, cyan, yelow and blue.
   c) Standard webcam Phillips SPC 900NC/00.
2. Experiment: From each image (recorded in a MPEG file) we find the SF2 images, and there we find the markers.

In figure 5 we have thre images from the described scenario. The ones on the left are the closest ones to the camera, the ones on the right are the farthest ones. Notice variations in illumination along the corridor. In figure 3 we show the SF2 images as follows: left corresponding to the middle one in figure 5 , middle after the analysis of the intensity and to the right a zoom of the previous one, showing that one mark is missing. In table 1 we show the detections performed on each mark, where 'x' means good detection and '+' incomplete detection.

Fig. 2 Natural images

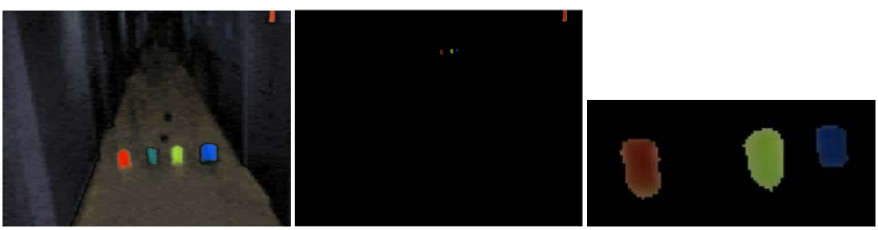


Fig. 3 SF2 images

Table 1 Measures

Milestone	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Distance in meters	2,6	4	6	8.4	10.8	12.8	14.5	17.3	20.7	26.2	31.7	36	41.9	46	50
Label 1	x	x	x	x	x	x	x	x	x	x		x	+	x	
Label 2	x	x	x	x	x	x	+								
Label 3	x	x	x	x	x	x	x	x	x	x	x	x	+	X	
Label 4	x	x	x	x	x	x	x	x	x	+					

4.2 Synthetic Images

The above color transformation has been applied to natural and synthetic images. Synthetic images have the advantage that we know with precision the color and geometry of the surface, as well as the illumination color. In figure 4 we show some of these images, in the top row we place the original image and on the bottom the computed SF2 images. First image is a monochromatic image, with a green surface. The second is a Voronoi tesselated surface painted with random colors. Last image is a bichromatic oval. We observe that SF2 images remove completely all the reflections, cancelling the specular component. In the Voronoi tesselated ring surface, besides canceling brights spots, colors have been enhanced.

The SF2 method has been ideated for robotic contexts. In figure 5 we show results on three natural images. The two first ones are customary marks in the previously described experiement, and the last one is used by other researchers in the literature of specular correction. The first two scenes show the magnification of the markers in the image. In the last case we see that the bright spots are cleanly removed, respecting original color.

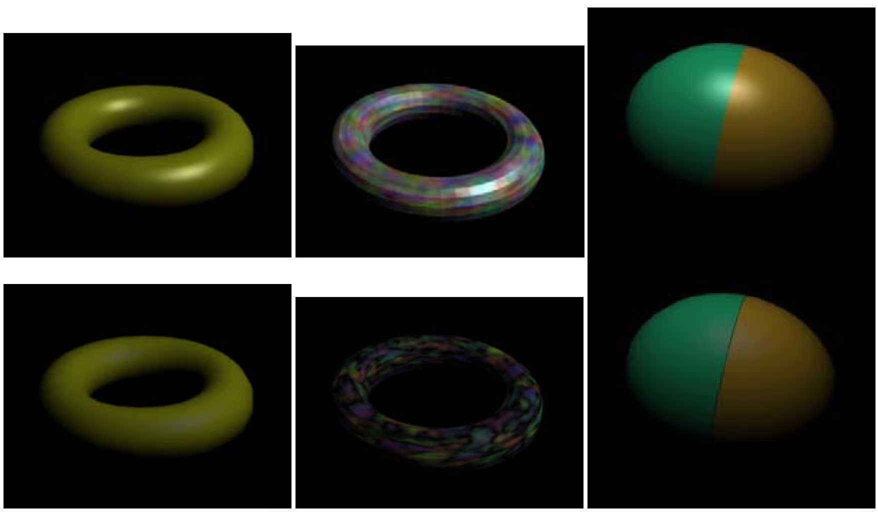


Fig. 4 Synthetic images

4.3 Real Robot Detection

The last experiment is the detection of small robots (SR1) in a real scene and real time. The robots are yellow color against a yelowist background, making visual detection tricky. The floor is very bright with many bright spots fromt above illumination. Besides, robot's upper part contains the printed board and some fixing for the cable being carried. The robots have lots of shadows, thus only a small part

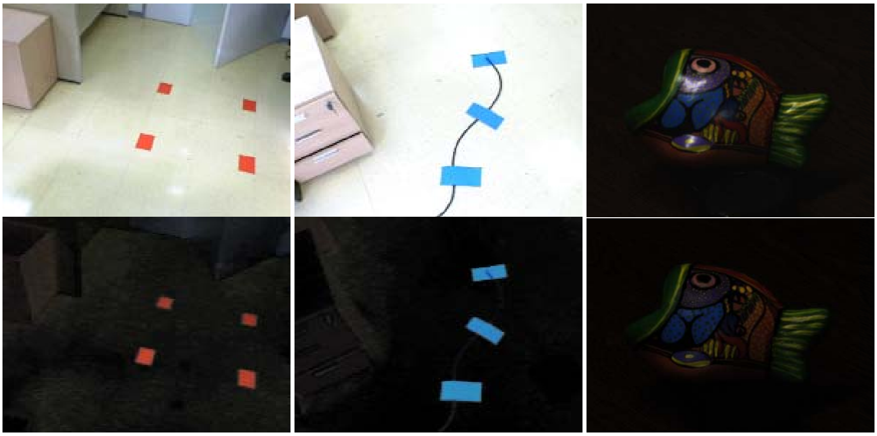


Fig. 5 Natural images

of the robot can be clearly detected as pure yellow. Figure 6 contains three images: first the capture from the scene, second its SF2 image, third the SF2 image intensity analysis to detect the robots. The web address http://www.ehu.es/ccwintco/index.php/SMC contains the original video. We must point out that illumination is not constant, there are doors, windows, etc.

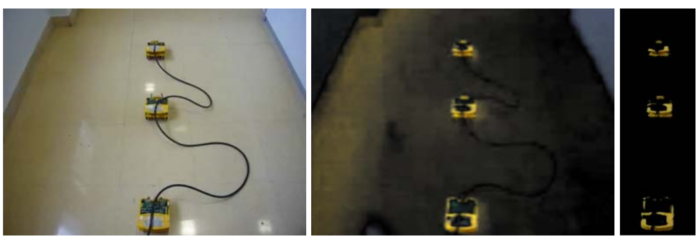


Fig. 6 Robots detection

5 Conclusions and Further Work

The work presented here proposes a method for color detection in images, characterized by:

1. Being fast and efficient.
2. Removes the specular component.
3. Magnifies color, preserving scene chromaticity, modifying only the intensity.
4. Can work in real time.

Other methods for the removal of the specular component are based on iterative methods that render them unsuitable for real time processing. In the future we will work on the color constancy problem and the color edge detection from the DRM point of view.

References

1. Woods, R.E., Gonzalez, R.C.: Digital Image Processing. Prentice Hall, Englewood Cliffs (2007)
2. Shafer, S.A.: Using color to separate reflection components. Color Research and Aplications 10, 43–51 (1984)
3. Tan, R.T., Ikeuchi, K.: Separating reflection components of textured surfaces using a single image. In: Proceedings of Ninth IEEE International Conference on Computer Vision, October 13-16, vol. 2, pp. 870–877 (2003)
4. Tan, T.T., Nishino, K., Ikeuchi, K.: Illumination chromaticity estimation using inverse-intensity chromaticity space. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, June 18-20, vol. 1, pp. I-673–I-680 (2003)
5. Yoon, K.-J., Chofi, Y.J., Kweon, I.-S.: Dichromatic-based color constancy using dichromatic slope and dichromatic line space. In: IEEE International Conference on Image Processing, ICIP 2005, September 11-14, vol. 3, pp. III-960–III-963 (2005)
6. Yoon, K.-J., Choi, Y., Kweon, I.S.: Fast separation of reflection components using a specularity-invariant image representation. In: IEEE International Conference on Image Processing, October 8-11, pp. 973–976 (2006); Bratman, M.E., Israel, D., Pollack, M.E.: Plans and resource-bounded practical reasoning. Computational Intelligence 4, 349–355 (1988)

Low Quality Data Management for Optimising Energy Efficiency in Distributed Agents

Jose R. Villar, Enrique de la Cal, and Javier Sedano

Abstract. Energy efficiency represents one of the main challenges in the engineering field. The benefit of the energy efficiency is twofold: the reduction of the cost owing to the energy consumption and the reduction in the energy consumption due to a better design minimising the energy losses. This is particularly true in real world processes in the industry or in business, where the elements involved may be considered as distributed agents. Moreover, in some fields like building management systems the data are full of noise and biases, and the emergence of new technologies -as the ambient intelligence can be- degrades the quality data introducing linguistic values. In this contribution we propose the use of the novel genetic fuzzy system approach to obtain classifiers and models able to manage low quality data to improve the energy efficiency in intelligent distributed systems. We will introduce the problem and some of the challenging fields are to be detailed. Finally, a brief review of methods considering the low quality data is related.

Keywords: Genetic Fuzzy Systems, Low quality data, Energy Efficiency, Building Automation.

1 Introduction

Analysing and optimising energy efficiency in processes has shown a promising field in different areas such as electric energy distribution [18, 29], energy saving

Jose R. Villar
University of Oviedo, Campus de Viesques s/n 33204 Gijón, Spain
e-mail: villarjose@uniovi.es

Enrique de la Cal
University of Oviedo, Campus de Viesques s/n 33204 Gijón, Spain
e-mail: delacal@uniovi.es

Javier Sedano
Instituto Tecnológico de Castilla y León, Lopez Bravo 70,
Pol. Ind. Villalonquéjar 09001 Buros, Spain
e-mail: javier.sedano@itcl.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 673–680.
springerlink.com

and sharing [5], efficient design and operation [2, 14], modeling and simulation [6], etc. In these areas, the main problem varies from how to fulfil the electric energy demand with the lowest cost or with the minimum line losses [18] to obtaining better models and controllers in order to better simulate or share the energy between distributed devices [6]. In general, the energy efficiency can take advantage of the multi-agent architecture, distributing the control and the optimisation decisions among all the intelligent devices [3, 27]. In what follows, the field related with daylight dimming control in building management systems- is to be analysed for the sake of simplicity, although the main conclusions can be extended to any other area.

Building Management Systems (BMS), also known as Building Automation Systems, are responsible for integrating all the automated systems in a building like the heating and ventilation automated control systems (HVACS), or the lighting and dimming control, etc. In a lighting control system (see Fig. 1), the lighting system controller is the software responsible for co-ordinating the different islands and of integrating the information from the BMS. In each island, a controller establishes the operation conditions of all the controlled ballasts according to the sensor measurements and the operation conditions given by the lighting system controller.

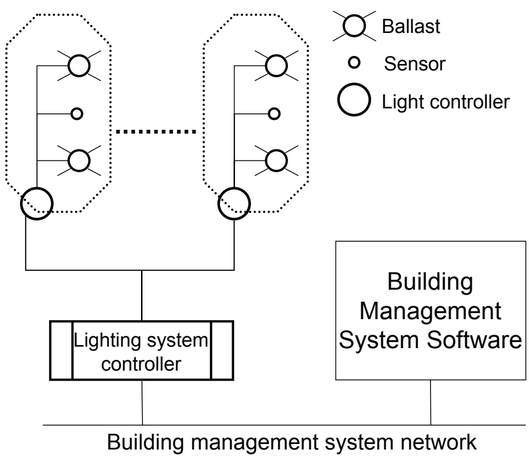


Fig. 1 The schema of a lighting control system. Each island includes a closed loop controller with the controlled gears, the luminosity sensors, the presence sensors, etc. The lighting system controller is responsible for the integration of all the islands.

In the literature of lighting control systems, the development of wireless sensor networks represents the main area of research [7, 17], introducing distributed sensors and deploying them, i.e., through web services. Researches related with simulation issues [6], sensor processing and data improvement [8], the effect of daylight in the energy efficiency [11] are also studied in depth in the literature. Moreover, the improvement in the energy efficiency and its measurement have been analysed in [11, 14, 16, 19].

Nevertheless, there is a lack in the use of all the information within the data gathered from the processes or the data sets. This meta-information is related with the low quality data due not only to the non stochastic noise or the precision of the sensors, but also due to the emergence of new technologies such as ambient intelligence and user profiles, which could be linguistic information. Genetic fuzzy systems (GFS) have been used in optimising energy saving and sharing in HVACS [1], but to our knowledge no GFS have been applied in lighting control systems. In our opinion, the use of GFS could improve the issues related with energy sharing and efficiency in distributed systems. We propose using the GFS able to deal with the meta-information to achieve a better the decision making process in the energy sharing among distributed agents -hence obtaining a better energy efficiency- and in the design of energy efficiency measures considering the low quality data.

In this research we propose the use of GFS for improving the energy efficiency in distributed systems taking advantage of the meta-data due to low quality data. We introduce a novel method for learning GFS with low quality data and then we propose how these methods can be deployed. We propose that a better decision making process and modeling through GFS can be carried out if the uncertainty in the data is managed. The remainder of this manuscript is as follows. First, a review of the literature concerned with considering the low quality data in modeling and in designing indexes is shown. Then, several different proposals for including low quality data are detailed. Finally, conclusions and future work are presented.

2 Issues in Low Quality Data Management

The need for algorithms able to face low quality data is a well-known fact in the literature. Several studies have presented the decrease in the performance of crisp algorithms as uncertainty in data increases [9, 25].

On the other hand, [15] analyses the complexity nature of the data sets in order to choose the better Fuzzy Rule Based System. Several measures are proposed to deal with the complexity of the data sets and the Ishibuchi fuzzy hybrid genetic machine learning method is used to test the validity of the measures. This research also concludes in the need to extend the proposed measures to deal with low quality data.

With low quality data we refer to the data sampled in presence of non stochastic noise or obtained with imprecise sensors. It is worth noting that all the sensors and industrial instrumentation can be regarded as low quality data. In our opinion, one of the most successful researches in soft computing dealing with low quality data is detailed in [4, 21]. In these works the mathematical basis for designing vague data awareness genetic fuzzy systems -both classifiers and models- is shown. The low quality data are assumed as fuzzy data, where each $\alpha-$cut represents an interval value for each data.

Finally, it is worth pointing out that the fitness functions to train classifiers and models are also fuzzy valued functions when faced with low quality data. Hence the learning algorithms should be adapted to such fitness functions [24].

The ideas and principles previously shown have been used in several applications with low quality data, with both realistic and real world data sets. [22] deals with the learning of fuzzy rule-based models with backfitting algorithms when the presence of missing data and discrepancies between the truth and the sampled data are included in the data set. Moreover, a filter-type feature selection, based on the Battiti mutual information feature selection method, is detailed in [23]. In this research a fuzzy extension of the mutual information measure is shown. A real world application is presented in [28], where the GPS measurements are considered inherently fuzzy data. A GFS is used to filter measurements and, thus, obtain a lower upper bound for the trajectory of a vehicle.

3 Enhancing the Energy Efficiency in Distributed Agents with Low Quality Data Management

As introduced in the previous sections, the use of the meta-information included in the data could improve the energy efficiency in processes, specifically, when the process is distributed among agents. The meta-information includes not only measurements -i.e., measurements from sensors- but also the precision of the sensors, their calibration, the non stochastic noise, etc. It is worth noting that non stochastic noise is the typical noise found in real world processes. On the other hand, multi-agent systems have been successfully employed in improving the energy efficiency in heating systems [20, 26], but always dealing with crisp data. We propose the use of GFS able to deal with quality data to enhance the energy efficiency in distributed agents. Three different examples of future work are outlined below.

3.1 Obtaining Models for Simulation of Lighting Systems

Simulation of lighting systems has been widely studied, some of them to improve the energy efficiency [6, 11, 19], while others are related to design comfort environments [13]. A lighting system simulation needs to analyse the light in a room to avoid the dark zones. A simulation will use models to estimate the values of the light any ballast produces, the output from the light sensors, etc. To our knowledge, no model has been obtained including the meta-information due to low quality data and, thus, the effect of the daylight and other variables are introduced artificially -i.e., by including such information within the input data set.

In our opinion, the use of GFS to obtain models for simulation of lighting systems would help in the integration of the meta-information. The use of GFS allows determining behaviour laws and interpretability of the phenomena. Moreover, if low quality data is included in obtaining the models of the lights, the sensors and the controllers, then the inputs to the models are not crisp data but low quality data. Therefore, the models should consider the ideas outlined in the previous section. Specifically, the GFS learning algorithms must deal with fuzzy inputs and non-crisp fitness functions, the fitness functions must be fuzzy valued functions.

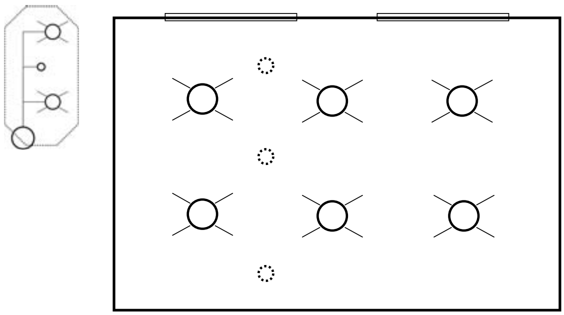


Fig. 2 The lighting system to simulate. Different places for the light sensor are proposed. The light measured will differ from one case to another.

Let us consider one simple case. Let us suppose the simulation of the lighting system shown in Fig. 2. It is clear that the shorter the distance to the windows the higher the daylight influence in the light measurements. The daylight should be estimated when no daylight sensors are available, from the inner light sensors,which are also influenced by the gears. Discriminating the daylight will be better when the meta-information from each sensor is considered than when it is not. Finally, we can introduce the multi-agent theory in the simulation design in order to represent the diversity of islands in the system. In the case of the example, each luminary, sensor and controller can act as an agent, behaving independently from each other.

3.2 Improving the Control Decisions in Lighting

As in the former case, several studies can be found in the literature concerning with the lighting control systems. It can be said that the luminance is the controlled variable when control loops are studied [14], or that the energy efficiency is the variable to be optimised when BMS are analysed [10]. To the best of our knowledge, the study of lighting control systems does not consider both variables at the same time.

We assert that the control decisions in lighting systems can take advantage of the meta-information aware GFS and of the multi-objective techniques to develop controllers. The use of low quality data and fuzzy fitness functions in their learning will produce more general case knowledge bases, so they can be applied in different scenarios; that is, the controllers are to be more robust. Moreover, the performance of lighting control systems with more than one objective should also be analysed. In fact, including more control variables could in practice penalise the control action response by introducing delays or steady state errors: the benefit is that the control action can afford a more energy efficient system. Finally, it is interesting to mention that energy efficiency measures can be extended to include meta-information. These extended measures can then be used as fitness functions in the learning of GFS.

3.3 Computer Assisted Decision Making

The more the information given the better the decisions are. As stated above, the meta-information has not been considered in the design of computer aided decision making assistants. On the other hand, the decision making process can be carried out by means of a GFS; the interpretability and robustness of the decision rules -the GFS knowledge base- represents a challenge in the literature. It is worth noting that in the decision making process the variables can be of numeric, crisp or linguistic types. Examples of each type are the measurements from sensors, the different levels obtained through thresholds or a set of possible values from human-machine interfaces; all of this information could be noisy or ambiguous.

We propose to include the meta-information -in the form of low quality data- in the learning of the knowledge bases of the GFS. The objective is twofold. On the one hand, the knowledge base will be robust due to the awareness of the noise in the numeric and crisp variables. On the other hand, as it will consider linguistic variables and their interpretability, will be made easier the integration of emergent technologies such as the ambient technologies.

Lastly, let us consider again the lighting control system shown in Fig. 1. If the energy efficiency in the whole system must be enhanced, then the energy efficiency in each island should also be accomplished. But as there are some objectives and restrictions fixed in each island, the problem can be viewed as a multi-agent system problem for which the energy efficiency must be improved on behalf of the management system while each agent should optimise its own objectives. In this case, we show that the three examples given in this section are highly related.

4 Conclusions

Improving the energy efficiency represents a challenge in the real world applications, especially distributed systems within building management systems. The higher the human interaction in field the higher relevance of intelligent techniques that consider the meta-information and interpretability. Meta-information refers to the information that is present in a process but rarely considered, such as the data non-stochastic noise or the sensor precision and calibration, but also the ambiguity in the linguistic and crisp data. Meta-information can be presented as low quality data.

GFS have been found valid in energy efficiency, but have also been extended to manage interval and fuzzy data. Interval and fuzzy data are mechanisms to represent the low quality data. In this work the extended GFS to manage fuzzy data is proposed to be used in the energy efficiency improvement. To illustrate the idea, three different cases are outlined, which represent the future work to be developed. We expect that fuzzy data awareness GFS will outperform the modeling and simulation process, but also the energy issues in distributed agents.

Acknowledgements. This research work is been funded by Gonzalez Soriano, S.A. by means of the the CN-08-028-IE07-60 FICYT research project and by Spanish M. of Science and Technology, under the grant TIN2008-06681-C06-04.

References

1. Alcalá, R., Alcalá-Fdez, J., Gacto, M.J.: Improving fuzzy logic controllers obtained by experts: a case study in HVAC systems. Applied Intelligence 31, 15–30 (2009)
2. Bernal-Agustín, J.L., Dufo-López, R.: Techno-economical optimization of the production of hydrogen from PV-Wind systems connected to the electrical grid. Renewable Energy 35(4), 747–758 (2010)
3. Booy, D., Liu, K., Qiao, B., Guy, C.: A semiotic multi-agent system for intelligent building control. In: Ambi-Sys 2008: Proceedings of the 1st international conference on Ambient media and systems (2008)
4. Couso, I., Sánchez, L.: Higher order models for fuzzy random variables. Fuzzy Sets and Systems 159, 237–258 (2008)
5. Davidsson, P., Boman, M.: Distributed monitoring and control of office buildings by embedded agents. Information Sciences 171, 293–307 (2005)
6. de Keyser, R., Ionescu, C.: Modelling and simulation of a lighting control system. Simulation Modelling Practice and Theory (2009), doi:10.1016/j.simpat.2009.10.003
7. Díaz, J., Rodríguez, E., Hurtado, L., Cacique, H., Ramírez, A., Vázquez, N.: LightNet a Reliable Option for Lighting Applications. In: Proceedings of the International Conference on Advances in Electronics and Micro-electronics, pp. 156–164. IEEE Computer Society, Los Alamitos (2008)
8. Doulos, L., Tsangrassoulis, A., Topalis, F.V.: The role of spectral response of photosensors in daylight responsive systems. Energy and Buildings 40(4), 588–599 (2008)
9. Folleco, A.A., Khoshgoftaar, T.M., Van Hulse, J., Napolitano, A.: Identifying Learners Robust to Low Quality Data. Informatica 33, 245–259 (2009)
10. Gligor, A., Grif, H., Oltean, S.: Considerations on an Intelligent Buildings Management System for an Optimized Energy Consumption. In: Proceedings of the IEEE Conference on Automation, Quality and Testing, Robotics (2006)
11. Hviid, C.A., Nielsen, T.R., Svendsen, S.: Simple tool to evaluate the impact of daylight on building energy consumption. Solar Energy (2009), doi:10.1016/j.solener.2008.03.001
12. Houwing, M., Ajah, A.N., Heijnen, P.W., Bouwmans, I., Herder, P.M.: Uncertainties in the design and operation of distributed energy resources: The case of micro-CHP systems. Energy 33(10), 1518–1536 (2008)
13. Kim, S.Y., Kim, J.J.: Influence of light fluctuation on occupant visual perception. Building and Environment 42(8), 2888–2899 (2007)
14. Li, D.H.W., Cheung, K.L., Wong, S.L., Lam, T.N.T.: An analysis of energy-efficient light fittings and lighting controls. Applied Energy 87(2), 558–567 (2010)
15. Luengo, J., Herrera, F.: Domains of competence of fuzzy rule based classification systems with data complexity measures: A case of study using a fuzzy hybrid genetic based machine learning method. Fuzzy Sets and Systems 161, 3–19 (2010)
16. Mady, A., Boubekeur, M., Provan, G.: Compositional Model-Driven Design of Embedded Code for Energy-Efficient Buildings. In: Proceedings of the 2009 IEEE International Conference on Industrial Informatics, pp. 250–255 (2009)
17. Malastras, A., Asgari, A.H., Baugé, T.: Web Enabled Wireless Sensor Networks for Facilities Management. IEEE Systems Journal 2(4), 500–512 (2008)

18. Martín, J.A., Gil, A.J.: A new heuristic approach for distribution systems loss reduction. Electric Power Systems Research 78(11), 1953–1958 (2008)
19. Park, T.J., Hong, S.H.: Development of an Experimental Model of BACnet-based Lighting Control System. In: Proceedings of the 2006 IEEE International Conference on Industrial Informatics, pp. 114–119 (2006)
20. Qiao, B., Liu, K., Guy, C.: A Multi-Agent System for Building Control. In: IAT 2006: Proceedings of the IEEE/WIC/ACM international conference on Intelligent Agent Technology, pp. 653–659. IEEE Computer Society, Los Alamitos (2006)
21. Sánchez, L., Couso, I.: Advocating the Use of Imprecisely Observed Data in Genetic Fuzzy Systems. IEEE Transactions on Fuzzy Systems 15(4), 551–562 (2007)
22. Sánchez, L., Otero, J.: Learning Fuzzy Linguistic Models from Low Quality Data by Genetic Algorithms. In: Proceedings of the IEEE Internacional Conference on Fuzzy Systems FUZZ-IEEE 2007 (2007)
23. Sánchez, L., Suárez, M.R., Villar, J.R., Couso, I.: Mutual Information-based Feature Selection and Fuzzy Discretization of Vague Data. Internacional Journal of Approximate Reasoning 49, 607–622 (2008)
24. Sánchez, L., Couso, I., Casillas, J.: Genetic Learning of Fuzzy Rules based on Low Quality Data. Fuzzy Sets and Systems (2009)
25. Van Hulse, J., Khoshgoftaar, T.M.: A comprehensive empirical evaluation of missing value imputation in noisy software measurement data. The Journal of Systems and Software 81, 691–708 (2008)
26. Villar, J.R., Pérez, R., de la Cal, E., Sedano, J.: Efficiency in Electrical Heating Systems: An MAS real World Application. In: Proceedings of the 7th International Conference on Practical Applications of Agents and Multi-Agent Systems (PAAMS 2009). LNCS, vol. 55, pp. 460–469. Springer, Heidelberg (2009)
27. Villar, J.R., de la Cal, E., Sedano, J.: A fuzzy logic based efficient energy saving approach for domestic heating systems. Integrated Computer-Aided Engineering 16(2), 151–164 (2007)
28. Villar, J.R., Otero, A., Otero, J., Sánchez, L.: Taximeter verification with GPS and Soft Computing Techniques. Soft Computing 14(4), 405–418 (2010)
29. Zhu, Y., Tomsovic, K.: Optimal distribution power flow for systems with distributed energy resources. International Journal of Electrical Power & Energy Systems 29(3), 260–267 (2007)

Interactive Multimedia Tabletops (IMT) for Casual Users

Andoni Beristain Iraola and Manuel Graña Romay

Abstract. In this paper Interactive Multimedia Tabletops are introduced, commonly denoted as tabletops. This natural Human Computer Interface uses the analogy of a conventional table, in order to permit collaborative work in the same physical space. Software applications for tabletops target two very different kind of scenarios and users. On one hand simple tasks oriented to casual users, and on the other hand more complex tasks oriented to professional environments. This paper focuses on tabletops designed for casual users from the physical aspect to the software applications.

1 Introduction

The current trend on computing devices and digital communication means is to blend in the environment, which is related to the concept of system ubiquity [25]. Human Computer Interaction (HCI) methods have remained almost unalterable in the last twenty years, and also the configuration of CPU box, screen, keyboard and mouse has not changed. The Interactive Multimedia Tabletop configuration is proposed as a step with much improved ergonomics for certain applications, mainly for two interaction paradigms: first the management of multimedia data, including Content Based Information Retrieval (CBIR) [3], organization and visualization and editing; and then the collective work by groups of people, sharing and manipulating multimedia information simultaneously.

An Interactive Multimedia Tabletop, commonly known as tabletop is the computer interface to an underlying computer system. It has the external appearance of a table shaped furniture, with some kind of video display on its

Andoni Beristain Iraola · Manuel Graña Romay
Grupo de Inteligencia Computacional (GIC), Universidad del País Vasco,
P° Manuel Lardizabal 1 - 20018 - Donostia, Gipuzkoa, Spain
e-mail: beristainandoni@yahoo.es, ccpgrrom@si.ehu.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 681–688.
springerlink.com

horizontal surface, where the graphical user interface (GUI) or Tangible User Interface (TUI) [5, 10, 12] is presented. Users interact with the virtual objects on the GUI/TUI by means of natural interaction methods like speech, hand gestures, multitouch or physical objects. Some systems are also multimodal. Following the idea of ubiquity, multimedia plays an important role on the aesthetics of the GUI/TUI, in order to distinguish Tabletops from ordinary computers. Tabletops are also focused on the visualization and manipulation of multimedia files, like videos, pictures, 3D, animations and sound.

Applications in Tabletops are either oriented to casual users, where simplicity of the interaction is more important than the language richness, or oriented to expert users, which do an intensive use of the system and an initial learning stage is feasible.

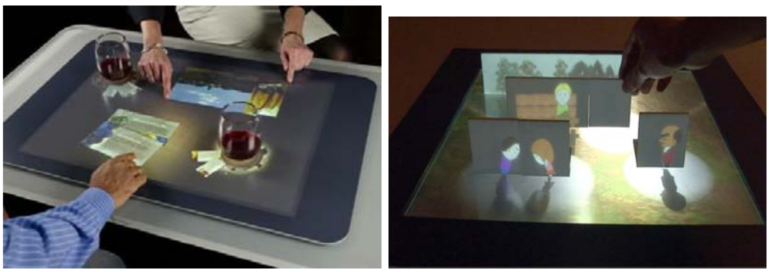


Fig. 1 Some tabletop systems

Tabletops provide several advantages over traditional computer systems. Tabletops allow users to interact face to face, acting as a mediator for communication and sharing data items, promoting a cooperative working environment for the completion of a common task [1, 19], both locally but also with other teams geographically distant [23, 17], thus, IMT are not restricted to small groups nor isolated environments.

2 Review

Research on Interactive Multimedia Tabletops is quite recent, for several reasons: (a) because multimedia data management was not generalized until the last half of the 90s[1] decade, (b) increased computational complexity implied by the need of natural interaction, and (c) the recent appearance of key technologies to support it, e.g. multitouch surfaces.

An example of systems designed in the 1980s decade is the VIDEOPLACE [14], which consists of a vertical screen where objects were projected, and the silhouette of the users was captured using a camera. The horizontal version of VIDEOPLACE was called VIDEODESK.

[1] http://en.wikipedia.org/wiki/Computer_graphics

Developed in the early 90s, DigitalDesk [26] consists of a physical desk with a camera and projector over it. The system projects images on the desk and was able to detect real objects, perform OCR, and recognize some hand gestures realized over the desk. Its design included the possibility to perform network interactions. The VideoDraw and Video Whiteboard [21, 22] allowed collaborative drawing among different users using virtual blackboards. The TeamWorkStation [11] combined the concepts of real and virtual desk. The DOLPHIN system [20] was composed of a vertical blackboard and an optical pencil to interact with it, which is able to interpret the strokes made to interact with the virtual objects in the blackboard graphical interface. Another system that made use of the stylus as interaction method was described in [4]. Early use of physical actuators for the interaction was demonstrated in the Active Desk [6], where "bricks" were used in a painting application.

Most of the recent IMT designs involve multitouch interaction. The most common multitouch techniques are based on Frustrated Total Internal Reflection (FTIR) and Capacitive Effects. FTIR are usually used in combination with one or more video projectors for display, while the Capacitive Effect is mainly used in systems whose displays are LCD screens.

Hand gesture recognition, is scarcely proposed as a mean of interaction in the latest systems. On the contrary, physical actuators are used by many systems. Moreover, in some systems the tabletop itself can even move the actuator [16].

Most of the latest research works about IMT focus their efforts on GUI improvement and the adaptation of current conventional software to the IMT paradigm, aiming towards simultaneous interaction between multiple users, locally and/or remotely [12, 23]. These works are not innovative on the technical aspects of the interaction method chosen.

Voice interaction is limited to a handful of research works. An interesting research work, [2] is an example of combination of voice and multitouch on an IMT, as well as an additional vertical screen that can be used to see remote users working in another connected IMT, by means of a camera integrated in the vertical screen.

3 IMT for Casual Users

This kind of IMT are focused on occasional interaction by any kind of user, in environments of great flow of people. Therefore, naturalness and simplicity of use are the most important design features. These requirements include the use of interaction methods and languages which are easy to learn or they do not even require any kind of learning and are also difficult to forget. In [8] several IMT from nine exhibitions are reviewed.

3.1 Perception of Naturalness Is Not Universal

The most important factor when designing the interaction method for these IMT kind is naturalness, but this feature rarelyis covered in the literature. Common sense gives people clues about how to interact with an object. This in fact means that several factors, namely the object external appearance, the object usage context, previous user interaction experiences and sociocultural aspects, make people feel an interaction method with a specific object more natural than others.

According to the Ecological Approach based on the work by Gibson [9], which defends that psychology should be the study of the interaction among people and with their environment, and its adaptation to the Human-Computer Interaction (HCI) context[18, 15, 7, 24, 27, 13], interaction naturalness depends on the object use context, and the interaction is determined by the *Ecological Constraints* and the *Affordances*. The *Ecological Constraints* are real world structures which guide people actions. And the *Affordances* are the objects attributes which permit people to know how to use them. Some people for example are reluctant to wear, grab or touch objects that have been previously used by many other people. And intrusive devices can also produce rejection, like fingerprint or eye recognition devices.

3.2 Interaction Languages for Casual Users

The interaction protocol followed with the IMT can be specified and treated as a language, we can define its lexicon, grammar and semantics. The most important aspects to have into account when defining an interaction language for an IMT focused on casual users those related to the ease with which the user can start interacting efficiently with the system: short learning time and persistence in the mind of the user (once learned, never forgotten). This is achieved using an intuitive and simple language which follows a simple logic without ambiguities.

Natural interaction languages are those with natural lexicon, grammar and semantics in the sense that they are immediately inspired in culturally accepted facts. These interaction languages include the voice communication using natural language. Of course, natural interaction languages are the most adequate for casual users. They can extrapolate their common life knowledge into the interaction with the system and the learning process can have the form of a game very easily. The combination with physical actuators can contribute with the haptic feeling to the naturalness of the interaction, but unfortunately the IMT GUI can not modify the actuators, except from very specific cases like [16].

Besides naturalness, simplicity is another design goal. Simplicity avoids overloading interaction elements (lexicon, grammar) with meanings, and thus avoids ambiguities. Users can accept and learn faster simple and clean interaction languages.

3.3 Guiding the User on the Interaction Possibilities

In the case of the casual users, the interaction capabilities perceived by the user must be complete, in the sense of being able to perform a successful interaction, from the first contact with the IMT. The main goal is to guide the user to interact with the IMT immediately, and give him clues to discover all the remaining interaction possibilities gradually so his interaction experience is as full as possible.

One strategy to perform such a user guidance, consists on the use of a short visual and optional tutorial, in the form of a game with increasing difficulty, showing part of the interaction possibilities available, and helping the user to develop his mental model of the IMT. Once the tutorial is finished, the user would be able to infer the rest of the possibilities. This is a kind of disguised training process, which should not be necessary for very natural interaction languages, but which in fact can be very effective in realistic environments.

Other strategy consists on letting the user interact freely with the IMT. Meanwhile, the user behavior is studied to offer suitable suggestions about how to fulfill the task the user is trying to accomplish, according to the IMT inference system. This user behavior analysis and modeling adds complexity to the underlying IMT computational intelligence system which performs pattern recognition, planning, and other intelligent tasks. Moreover, this kind of systems can be annoying to the user, because it may provide non required suggestions, be silent when needed, it can be intrusive in the normal work of the user.

The use of information posters around the IMT can provide a first context about the interaction possibilities available. Although this kind of help is useful to provide a first impression about the IMT, it is not enough, unless extensive information is provided. This approach reduces the naturalness of the interaction, and it is limited to very basic information. However, careful design of the information posters may be very attractive to the users, which is something valuable in the case of the casual users.

3.4 Interaction Naturalness Measures for Casual User IMT

It is necessary to have naturalness measures to evaluate in advanced the potential success possibilities of an IMT for casual users in terms of user acceptance. With that in mind, two measures are proposed.

- Time Period Required to Master the Interaction: It is defined as the time period required to achieve an efficient interaction, which means to acquire the ability to perform a task with the minimum resources possible. The shorter this time period is, the more natural the interaction method is. A comparison with conventional alternatives like the keyboard and mouse can be used as measure.

- IMT Occupancy Rate: It is a measure of the acceptation or rejection of an IMT by users. It is defined as the time period when the IMT has been in use divided by the time period when people has been in the proximity of the IMT. This measure ranges from zero to one and the bigger this value is, the bigger the acceptance of this IMT is.

4 Conclusions

Interactive Multimedia Tabletops are a new tool for collective work environments and also for information retrieval in public places. This two environments have very different requirements. While in work environments productivity and efficiency are the most important characteristics, in public places the most important is that interaction must be easy and enjoyable, and an special effort must be put in making the IMT appealing and avoiding any kind of rejection. Therefore a first distinction has been done between IMT focused on regular and casual users.

IMT make use of natural interaction methods, which usually involves the use of hand gestures, physical object manipulation and natural language, i.e. speech. Most of this kind of interaction methods have been reviewed in the context of IMT .

Aesthetics both in the GUI/TUI workspace and the hardware's external appearance receive a special attention, and ubiquity issues are also present. And multimedia data is the main kind of data manipulated on them. Everything is devoted to natural and easy interaction and to break the barrier between users and computers.

The two main design objectives for casual user focused tabletops are: being aesthetically appealing to maximize the acceptance of the system, avoiding any contempt, and the naturalness/simplicity of the interaction.

The new multiple simultaneous user interaction models are a challenge to the design of virtual object interaction and collaboration. There is still a lot of work to do in user group management and technical aspects both in hardware and software, but this is a research area which has received much interest.

References

1. Chatty, S., Lemort, A., Vales, S.: Multiple input support in a model-based interaction framework, pp. 179–186 (October 2007)
2. Chen, F., Eades, P., Epps, J., Lichman, S., Close, B., Hutterer, P., Takatsuka, M., Thomas, B., Wu, M.: Vicat: Visualisation and interaction on a collaborative access table. In: International Workshop on Horizontal Interactive Human-Computer Systems, pp. 59–62 (2006)
3. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image retrieval: Ideas, influences, and trends of the new age. ACM Comput. Surv. 40(2), 1–60 (2008)

4. Elrod, S., Bruce, R., Gold, R., Goldberg, D., Halasz, F., Janssen, W., Lee, D., McCall, K., Pedersen, E., Pier, K., Tang, J., Welch, B.: Liveboard: a large interactive display supporting group meetings, presentations, and remote collaboration. In: CHI 1992: Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 599–607. ACM, New York (1992)
5. Fishkin, K., Kenneth, P.: A taxonomy for and analysis of tangible interfaces. Personal Ubiquitous Comput. 8(5), 347–358 (2004)
6. Fitzmaurice, G.W., Ishii, H., Buxton, W.A.S.: Bricks: laying the foundations for graspable user interfaces. In: CHI 1995: Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 442–449. ACM Press/Addison-Wesley Publishing Co., New York (1995)
7. Gaver, W.W.: Technology affordances. In: SIGCHI 1991, pp. 79–84. ACM Press, New York (1991)
8. Geller, T.: Interactive tabletop exhibits in museums and galleries. IEEE Computer Graphics and Applications 26(5), 6–11 (2006)
9. Gibson, J.J.: The Ecological Approach to Visual Perception. Lawrence Erlbaum Associates, Hillsdale (1986)
10. Ishii, H., Ullmer, B.: Tangible bits: towards seamless interfaces between people, bits and atoms. In: CHI 1997: Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 234–241. ACM, New York (1997)
11. Ishii, H., Miyake, N.: Toward an open shared workspace: computer and video fusion approach of teamworkstation. Commun. ACM 34(12), 37–50 (1991)
12. Kim, M., Maher, M.: The impact of tangible user interfaces on spatial cognition during collaborative design. Design Studies 29(3), 222–253 (2008)
13. Kirsh, D.: The context of work. Human-Computer Interaction 16(2), 305–322 (2001)
14. Krueger, M.W., Gionfriddo, T., Hinrichsen, K.: Videoplace: An artificial reality. SIGCHI Bull. 16(4), 35–40 (1985)
15. Norman, D.A.: The Psychology of Everyday Things. Basic Books (April 1988)
16. Patten, J., Ishii, H.: Mechanical constraints as computational constraints in tabletop tangible interfaces. In: CHI 2007: Proceedings of the SIGCHI conference on Human factors in computing systems, pp. 809–818. ACM, New York (2007)
17. Pauchet, A., Coldefy, F., Lefebvre, L., Louis Dit Picard, S., Perron, L., Bouguet, A., Collobert, M., Guerin, J., Corvaisier, D.: Tabletops: Worthwhile experiences of collocated and remote collaboration, pp. 27–34 (October 2007)
18. Rasmussen, J., Rouse, W.B.: Human detection and diagnosis of system failures. Plenum Press, New York (1981)
19. Scott, S.D., Grant, K.D., Mandryk, R.L.: System guidelines for co-located, collaborative work on a tabletop display. In: European Conference Computer-Supported Cooperative Work, ECSCW 2003 (2003)
20. Streitz, N.A., Geisler, J., Haake, J.M., Hol, J.: DOLPHIN: Integrated meeting support across local and remote desktop environments and liveboards. In: Computer Supported Cooperative Work, pp. 345–358 (1994)
21. Tang, J.C., Minnenan, S.L.: Videodraw: A video interface for collaborative drawing. In: Proc. of CHI 1990, Seattle, WA, pp. 313–320 (1990)
22. Tang, J.C., Minneman, S.L.: Videodraw: a video interface for collaborative drawing. ACM Trans. Inf. Syst. 9(2), 170–184 (1991)
23. Tuddenham, P., Robinson, P.: Distributed tabletops: Supporting remote and mixed-presence tabletop collaboration, pp. 19–26 (October 2007)

24. Vicente, K.J.: A few implications of an ecological approach to human factors. In: Global Perspectives on the Ecology of Human-Machine Systems, pp. 54–67. L. Erlbaum Asso., Hillsdale (1995)
25. Weiser, M.: The computer for the 21st century. SIGMOBILE Mob. Comput. Commun. Rev. 3(3), 3–11 (1999)
26. Wellner, P.: Interacting with paper on the digitaldesk. Commun. ACM 36(7), 87–96 (1993)
27. Woods, D.: Toward a theoretical base for representation design in the computer medium: ecological perception and aiding cognition. In: Global Perspective on the Ecology of Human-Machine Systems, pp. 157–188. L. Erlbaum Associates, Mahwah (1995)

Emotion Elicitation Oriented to the Development of a Human Emotion Management System for People with Intellectual Disabilities

R. Martínez, K. López de Ipiña, E. Irigoyen, N. Asla, N. Garay,
A. Ezeiza, and I. Fajardo

Abstract. The integration of people with intellectual disabilities into working and social environments is one of the main issues in applying Information and Communication Technology (ICT) into Assistive Technology field. In order to solve this problem, the use of Intelligent Tutoring Systems (ITSs) is a fact among the attendance community. Human Emotion Management System (HEMS) enhances the performance of ITSs. In this work a HEMS is included in a new ITS system (LAGUNTXO) which adds user affective information. People with intellectual disabilities and elderly need physical and intellectual support. This may be achieved by a HEMS which attempts to solve critical situations, like those presented in blockage stages. In this paper, new methods and material to analyze and

R. Martínez · K. López de Ipiña · E. Irigoyen · A. Ezeiza
Departamento Ingeniería de Sistemas y Automática, Escuela Técnica superior de Ingeniería, Bilbao, Escuela Politécnica, Donostia, Grupo de Inteligencia Computacional
e-mail: {raquel.martinez,karmele.ipina,eloy.irigoyen, aitzol.ezeiza}@ehu.es

N. Asla
Departamento de Psicología Social y Metodología de las Ciencias del Comportamiento, de la Facultad de Psicología, Donostia
e-mail: nagore.asla@ehu.es

N. Garay
Departamento de Arquitectura y Tecnología de Computadores, Laboratorio de Interacción Persona Computador para Necesidades Especiales, Facultad de Informática
Universidad del País Vasco/Euskal Herriko Unibertsitatea
Plaza de Europa1, Donostia, 20008 Spain
e-mail: nestor.garay@ehu.es

I. Fajardo
Department of Developmental and Educational Psychology
University of Valencia, Av. Blasco Ibáñez, 13, 46010 Valencia, Spain
e-mail: infabra@uv.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 689–696.
springerlink.com

classify human emotions and corresponding elicited emotions have been tested. This study has provided finding emotional changes which give the opportunity of detecting blockage situations in people with disabilities. The evaluation results of the created elicitation material and methods indicate a high level of accuracy with respect to others classically presented in the literature.

Keywords: Intelligent Tutoring System, Human Emotion Management System, Human Emotion Elicitation Database.

1 Introduction

This paper presents new results in assessing a Human Emotion Management System (HEMS). In the framework of a research line about integrating people with disabilities into working and social environments, our research group is developing an Intelligent Tutoring Systems (ITSs) which includes a General Task Management System (GTMS) functionality to optimize the ITS performance. HEMS is included into the GTMS as a tool for analysis, evaluation and identification of human emotions.

The HEMS has been created in order to improve the ITS performance by measuring and adding user emotion information [1][2][3]. People with intellectual disabilities as the elderly people need physical and intellectual support. This may be achieved by a HEMS which attempts to solve critical situations, as those presented in blockage stages. Frequently, people with intellectual disabilities have blockage situations where they cannot take a decision or answer to a stimulus. Typically, these situations appear either when they are anxiously, or when they suffer a panic attack, or when the requirements are further than they can do. In many cases, the main objective is detecting emotional changes, instead of classifying emotions, in order to attend these situations before beginning.

In section 2, human emotion survey and analysis is presented. Next, human emotion testing methodologies and emotion experimentation profiles in an experimental stage frame are shown. In section 4 evaluation results of the created elicitation material and methods are presented. Finally, some concluding remarks and future outlines are pointed out.

2 Human Emotion Analysis

According to Cañamero [4], human emotion entails distinctive integrated ways of perceiving and assessing situations, processing information, and modulating and prioritizing actions. To this respect, emotions can be seen as different ‘cogni-tive modes’ that have a ‘global’ and synchronized influence in our perceptual, cognitive, bodily and behavioural relation with the external world. Achieving this in computational affective architectures involves a number of challenges, many of them largely unexplored, such as the next ones: Which aspects of cognition need to be placed in ITS architectures to be able to speak of ‘cognitive mode’? What mechanisms are required to implement different effects of various emotions? How can computational and assistive devices take into account cultural and individual differences in the synthesis of emotions as ‘cognitive modes’? These notions are closely related in multiple ways and different assistive device architectures have implemented them.

Emotion can be understood as “an impulse that induces the action”, causing automatic reaction behaviours to the environmental stimulus. From the point of view of Psychology, emotion is a feeling expressed by a physiologic function like facial reactions, cardiac pulse and behavioural reactions like blockage situations, aggressions, crying, etc. However, many times the individual masks their emotions as a result of a cultural learning process (“socialized emotions”). Specifically, people with intellectual disabilities cover their emotions by mean of unpredictable emotional stages. In such cases, the emotional socialized response does not correspond with the pure emotional response because the individual obscures the situations (predictable and non predictable environments).

2.1 Modelling Emotions Physiologically Perceivable

In a perceivable emotion physiological answers like facial reactions or pulse include conduct reactions like blockage situation, aggression, crying, etc. The use of automatic systems of direct measurement like physiological cameras, microphones or sensors will allow detecting and measuring emotion in a trustworthy way. Human interaction includes emotional information of the interlocutors that is transmitted explicitly through the language and implicitly through nonverbal communication. The nonverbal information, that frequently it is transmitted by means of corporal gestures, attitudes, face, modulations of the voice, expressions, etc., are of great importance in the human communication, since it has a great effect on the communicative disposition of the interlocutors and on the intelligibility of the speech. Nevertheless, it has been appraised that these characteristics that are associated to the interpersonal relations also appear in the communication with the computers. But, how can the computer detect the mood of the user? Lang et al. suggest that there are 3 systems implied in the expression of the emotions and could serve like indicators to detect the user emotions: Verbal information (user’s auto-report on the perceived emotion), Conduct (registry of face and gesture expressions and paralinguistic parameters of the voice), Psycho-physiological answers (registry of cardiac rate, conductance of the skin, electro-brain answer, etc.)[5].

Therefore, a computer based system which registers and recognizes user’s emotions will be necessary, to determine which basic emotions go away to measure and which are their subjective or verbal, related to conduct and psycho- physiological. For classifying emotions there is not an established model. According to some authors, human emotions are restricted to 6, although in some cases it is possible to find experimental results classifying more than 15 emotions. This may cause confusion in the development of an automatic emotion recognition system. Furthermore, while one would expect a set of basic emotions to be consistently recognized across cultures—in other words, being universal—evidence suggests that there is minimal universality at least in the recognition of emotions from facial expressions [2].

In this work, a preliminary experimental stage has been defined for classifying emotions where an adapted brief self-report Emotion Regulation Questionnaire with 12 emotions is used. In order to increase the participant perceptions this questionnaire enquires new emotions.

2.2 Modelling Emotions Physiologically Non-perceivable

When a person camouflages emotions due to cultural learning (socialized emotions), a perceptible physiologic response is not produced and the classical emotion measure system (cameras, microphones, lexical information, etc.) cannot be used, so a different measuring system should be developed, incorporating both individual artificial emotional patterns (emotional data bases of human emotional patterns) and emotional memories (data bases of human experiences). In these patterns, the individuality of the people and the cultural components have to be considered. The system of measurement in this case will be composed, on the one hand, by artificial emotional patterns of individuals, and on the other hand, by emotional memories, being conditional on its answer to measure the emotion in an automatic learning based on human brain.

Our ITS must be based on techniques of prediction, memorization and automatic learning of human experiences. The introduction of artificial emotional capabilities in ITS systems will clearly improve them providing higher robustness when interacting with the environment. Therefore, first step is to obtain appropriate human modelling and measurement of both perceptible and not perceptible emotions. In this work, Non Intrusive Intelligent Systems (NIIS) are created in order to measure biological signals in people with intellectual disabilities taking privacy and ethical issues into account.

3 Experimental Stage

Our most recent works are focused on the study and control of human reactions. This is a very intricate topic, since it involves human emotion detection, it requires a complex sensor configuration, and it needs many arduous experiments in order to achieve reliable results. In general, the law forbids any kind of experimentation with people with intellectual disabilities in order to protect these vulnerable persons. Besides, the results obtained with a set of volunteers are difficult to extrapolate to any kind of user, including people with intellectual disabilities. In consequence, the design of the tests must be carried out with insight and with the collaboration of experts in this topic.

Research is a profit for humanity by itself, and it contributes to the acquisition of new knowledge. Nevertheless, the possible benefits must be weighted up with the possible damages or abuses that might be committed while experimenting with human beings. The experiments must be carried out respectfully and with sheer consideration to the test users that participate in them. These restrictions are positive, but the law recommends some extra restrictions for experimentation with people with limited autonomy.

We would like to remark that despite everybody could benefit of the results of research on assistive technology, accessibility and intelligent environments, the participation of persons with disability in these research experiments are of lively interest, since these persons have many more difficulties of adaptation than other people, and they are more sensible to bad technological design. In the particular case of the people with intellectual disabilities, they also face a great risk of social exclusion, so any engineer involved in research related to persons must try to consider their point of view. They are interesting for the research community, and they might accept this request for help with glad, but the caution must prevail when asking them for cooperation in a project. The humanistic and ethical view is absolutely imperative. This ethical point of view should be complemented with legal considerations as well. For modelling human emotion, a previous study in laboratories producing emotion elicitation is relevant [4]. Thus, we have developed an elicitation video database based on video, and an experimental prototype of HEMS based on Machine Learning (ML) algorithms.

4 Preliminary Results

4.1 Elicitation Video Database

Human emotion is a widely studied field on neuronal process theory, as well as sociology, and cognitive process theory. Many emotion elicitation techniques have been studied: Mental image reproductions, displaying emotional categorized films, displaying categorized slides, real-life techniques, autobiographical memories, hypnosis, facial postures, Velten mood induction technique, music, among others.

Displaying emotional categorized films seems to achieve better results due to its dynamic profile [6]. Furthermore, it has been widely observed that emotional categorized films can elicit strong subjective and physiological changes. Moreover, the dynamic profile of this technique obtains an optimal artificial model of reality, conforming practical problems of real-life technique, and providing better procedures to comply with ethical issues. For that reasons, it seems to offer a very good technique to elicit emotion in laboratory, especially for specific emotional states.

Films are effective stimuli to elicit emotions. A set of films to elicit specific emotions was developed in a previous study by Gross et al [6]. Nowadays, this set of films is been used as valid material in current studies. In our work, some films were replaced for actual films. A first validation of these films (*Gross*) was made in order to test if those films were still valid in our society. After analyzing the results, we find that many of the films were not valid for different reasons: old films with bad imagine and sound quality; translation to our language was not effective; they were movies that many people saw and no longer come to move. We prepared a database with more than 50 films, consulting film experts, owners from video clubs, internet forums and also looked for real facts where they gave more emotions than a film. We consult the work made by Stephens [7] and Schaefer [8] for analysing the methodology in their experiments. We took 20 video clips from our database. Furthermore, it was necessary to place a neutral video between films, so each person can recover its emotional basal level.

The database was classified in 7 emotions. Subsequently, the films were mounted in three groups with 10 clips each one, where 3 of them repeated an used emotion. In order to organize appropriately the projection, it was necessary to avoid editing neither 2 clips with the same emotion, nor with the same valence [7]. Finally we obtained results with 45 volunteers and more than 1000 emotional answers. Each answered questionnaire related to one emotional film was composed by a set of 16 emotional descriptions and the 3 scales: amusement, contentment and valence.

We carried out a brief validation of the database in new experiments, against a subset of *Gross and Real Films* (*Gross+RF*). Figure 1 shows relevant conclusions in this validation. Analyzing the values related to Sadness, it can be observed that the video clips used in second set obtained better values in the emotional answer.

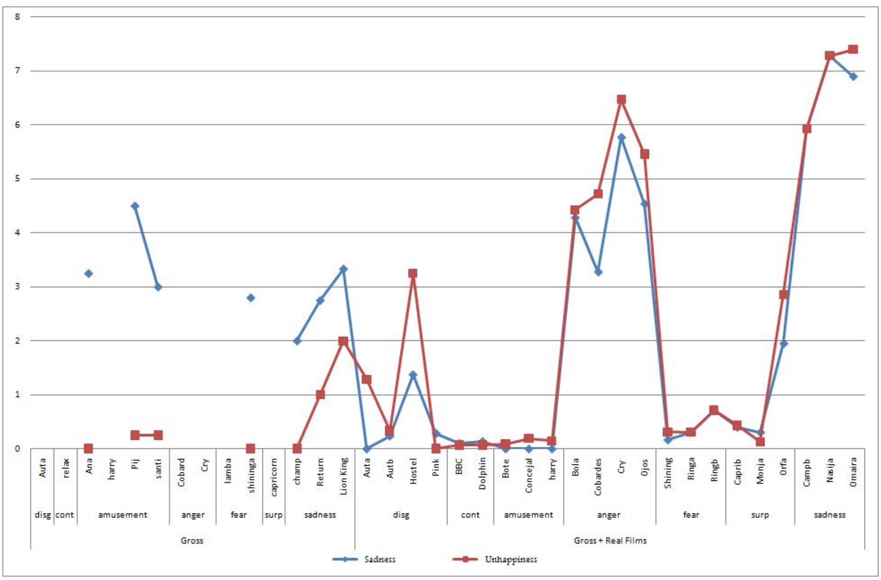


Fig. 1 Validation of Sadness over *Gross* and *Gross and Real Films* sets

Table 1 Description of the experiments

	Per Num	Film Num	Real Scene N	Instances
Gross	5	13	1	61
Gross+RF	7	7	2	62
Gross+RF-SS	45	30	6	562

4.2 Evaluation

In the supervised learning task, a classification problem has been defined where the main goal is to construct a model or a classifier able to manage the classification itself with acceptable accuracy, in order to classify the emotional answer. With this aim, some variables are used for identification of different elements, the so called predictor variables. The basic problem of Machine Learning is concerned with the induction of a model that classifies a given object into one of several known classes. In order to induce the classification model, each object is described by a pattern of ***d*** features. For the current problem, each sample (emotional answer) is composed by the set of 16 emotional descriptions, while the label value is one of the seven emotions identified. The single paradigms used in our HEMS which come from the family of Machine Learning (ML) are briefly introduced: C4.5, Multilayer Perceptron, Naive Bayes classifiers, KNN and Support Vector Machines.

The above mentioned methods have been applied over the cross-validated data sets. Each instance corresponds to an emotional answer. Three experiments were carried out (Table 1). *Gross* and *Gross+RF* have been described in 4.1 and *Gross+RF-SS* is based on a subset extract from *Gross+RF* with similar size to *Gross* that has been created in order to compare appropriately the results. Table 2 shows the classification results obtained using the whole set of variables, for *Gross*, *Gross+RF* and *Gross+RF-SS* respectively showing the total average for each classifier 10-fold cross-validation. Results show a good performance for most paradigms obtaining the best accuracy for SVM. On the other hand in *Gross+RF* and *Gross+RF-SS* the accuracies outperform the previous ones in more than 10%. It must also be highlighted that the new database improves the well classified rate for all the ML paradigms, as Table 2 shows.

Table 2 10-fold cross-validation accuracy *Gross*, *Gross+RF* and *Gross+RF-SS*

	Gross	*Gross+RF*	*Gross+RF-SS*
KNN	44,26	79,03	74,55
Multilayer Perceptron	77,04	79,23	74,33
NB	67,21	70,96	63,87
C4.5	63,93	69,35	69,57
SVM	73,77	83,87	78,46

5 Concluding Remarks and Future Outlines

A new Human Emotion Management System structure is been designed in order to introduce non-perceivable human emotions at the ITS. As a result of the work, a HEMS prototype has been developed. This management system will be integrated into a General Task Management System functionality which enhances our Intelligent Tutoring System performance. Despite everybody could benefit of the results of research on assistive technology, accessibility and intelligent environments, the participation of people with disabilities in these research experiments will be of lively interest, since these persons have many more difficulties of adaptation than other people, and they are more sensible to bad technological design.

In this work, new methods and material to analyze and classify human emotions, and corresponding elicited emotions have been tested. This study has provided finding emotional changes which give the opportunity of detecting blockage situations in people with disabilities. The evaluation results of the created elicitation material and methods indicate a high level of accuracy with respect to others classically presented in the literature. Finally, new studies are going to be performed in the future taking new affective resources into account.

Acknowledgements. The involved work has received financial support from the Spanish Ministerio de Ciencia y Tecnología, project TSI2006-14250-C02-01, and the Spanish Ministerio de Industria, Turismo y Comercio, project PDM-2006-178.

References

1. Cowie, R., Douglas-Cowie, E., Cox, C.: Beyond emotion archetypes: Databases for emotion modeling using neural networks. J. Neural Networks 18, 371–388 (2005)
2. Fragopanagos, N., Taylor, J.G.: Emotion recognition in human–computer interaction. J. Neural Networks 18, 389–405 (2005)
3. Taylor, J., et al.: Emotion and brain: Understanding emotions and modelling their recognition. J. Neural Networks 18, 313–316 (2005)
4. Cañamero, L.: Emotion understanding: From the perspective of autonomous robots research. J. Neural Networks 18, 445–455 (2005)
5. Lang, P.J., Öhman, A., Vaitl, D.: The International Affective Picture System (Photografic slides). University of Florida (eds.), Center for Research in Psychophysiology, Gainesville, FL (1998)
6. Gross, J., Levenson, W.: Emotion Elicitation Using Films. J. Cognition and Emotion 9, 87–108 (1995)
7. Stephens, C.L.: Autonomic Differentiation of Emotions: A Cluster Analysis Approach. Technical Report, Virginia Polytechnic Institute and State University (2007)
8. Schaefer, A., Nils, F., Sanchez, X., Philippot, P.: A multi-criteria assessment of emotional films. University of Louvain (eds.) Report for publication, Louvain-La-Neuve, Belgium (2005)

Acoustic Phonetic Decoding Oriented to Multilingual Speech Recognition in the Basque Context

N. Barroso, K. López de Ipiña, and A. Ezeiza

Abstract. The development of Large Vocabulary Continuous Speech Recognition systems involves issues as: Acoustic Phonetic Decoding, Language Modelling or the development of appropriated Language Resources. In the state of the art, new techniques for reusing Language Resources of more resourced related languages is becoming of great interest, and there is also a growing interest on Multilingual systems. This paper describes the initial experiments on multilingual recognition and cross-lingual adaptation carried out in order to create a robust Multilingual Speech Recognition system for the Basque context. The interest on Multilingual Systems arouses because there are three official languages in the Basque Country (Basque, Spanish, and French), and there is much linguistic interaction among them, even if Basque has very different roots than the other two languages.

Keywords: Automatic Speech Recognition, Multilingual System, Large Vocabulary Continuous Speech Recognition.

1 Introduction

Automatic Speech Recognition (ASR) is a broad research area that absorbs many efforts from the research community. Indeed, many applications related to ASR have progressed quickly in recent years, but these applications are generally very language-dependent. Therefore, the development of a robust system is a much tougher task for under-resourced languages, even if they count with powerful languages beside it. The development of Large Vocabulary Continuous Speech

---

N. Barroso
Irunweb Enterprise, Auzolan 2B – 2, Irun, 20303 Spain
e-mail: nora@irunweb.com

K. López de Ipiña · A. Ezeiza
Departamento Ingeniería de Sistemas y Automática, Grupo de Inteligencia Computacional
Escuela Politécnica Universidad del País Vasco/Euskal Herriko Unibertsitatea
Plaza de Europa1, Donostia, 20008 Spain
e-mail: {karmele.ipina,aitzol.ezeiza}@ehu.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 697–704.
springerlink.com

Recognition systems involves issues as: Acoustic Phonetic Decoding, Language Modelling or the development of appropriated Language Resources (figure 1).

Since the main goal of our project is the development of ASR systems in the Basque context, it was essential to create resources and tools for all the languages spoken by the potential users.

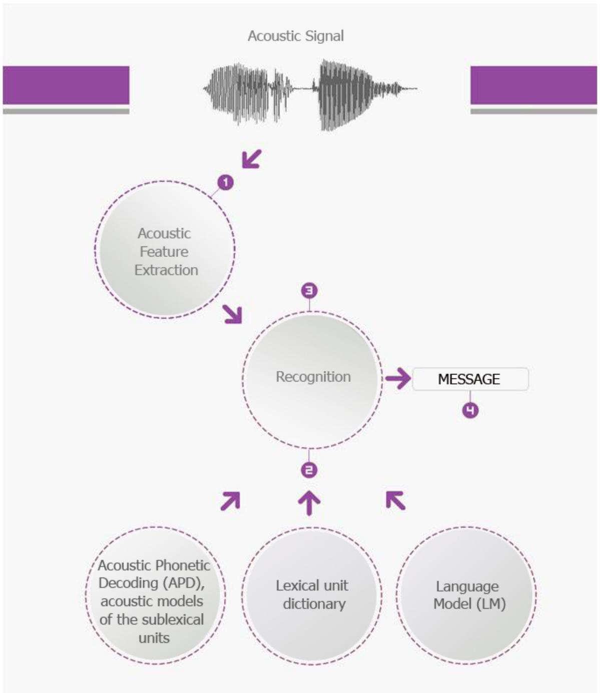


Fig. 1 Scheme of the process of Speech Recognition

Thus, there were many difficulties in the case of Spanish and French, but the hardest obstacle to overcome was the development of resources and tools for the multilingual system and for Basque. Nevertheless, there is growing interest in minority languages such as Basque, even taking into account the challenge that has to be faced in these cases. Many respectable ideas have been developed on Basque Speech Recognition [1], and nowadays there are some Language Resources developed for Basque [2] that encourage further research. And although there is much work to do with Basque Speech Recognition alone, the interest on Multilingual Systems arouses in the Basque Country because there are three official languages in use (Basque, Spanish, and French), and there is much cross-lingual interaction between them, even if Basque has very different roots than the other two languages. Indeed, the speakers tend to mix words and sentences in the three languages in their discourse, and the acoustic interactions between the three languages and between the Basque dialects are fairly interesting from the researchers' point of view.

One of the fields that have shown most interest in Automatic Speech Recognition is the Mass Media communication. Most of the mass media in Basque Country use Spanish, French, and/or Basque, and many of them have shown their interest in the development of Index Systems for their media. Thus, the three languages have to be taken into account to develop an efficient Speech Recognition system in this field. Many projects have been developed with several European languages [3], but Basque has fewer resources than many of them. In order to decrease the negative impact that the lack of resources has in this area, the alternative surges in the form of cross-lingual Acoustic Modelling [4]. Wheatley et al. [5] already suggested the idea to train phoneme models for a new language using other languages and they implemented a number of different metrics for measuring similarities among cross-language phonetic models. The idea behind cross-lingual speech recognition is to transfer existing resources of Acoustic Models to a target language without using a speech database in this target language [6]. This way the usage of a complete speech database in the target language can be avoided. Therefore, it seems an interesting alternative for Basque language, and this paper tackles this issue developing a baseline Acoustic-Phonetic Decoder for each of the three languages in order to compare them and to facilitate further experiments on cross-lingual Acoustic Modelling.

Therefore our first goal to deal with the development of a Multilingual Large Vocabulary Continuous Speech Recognition (MLVCSR) system in Basque context will be create a robust Acoustic Phonetic Decoding (APD) system based on the allophone set of the three languages (Spanish, French and Basque). This system will decode the acoustic signal in series of allophonic acoustic units.

2 Phonetic Features of the Languages

The analysis of the features of the languages chosen is a crucial issue because they have a clear influence on both the performance of the APD and on the vocabulary size of the system. In order to develop the APD, an inventory of the sounds of each language was necessary. Table 1 summarises the sound inventories for the three languages expressed in the SAMPA notation. Each sound would be taken into account in the phonetic transcription tools used in the training process.

In order to get an insight of the phonemes system of these three languages, we would like to remark some of the features mentioned above. In the one hand, Basque and Spanish have very similar vowels if not the same. The Basque language itself has many odd occurrences of other vocals, but many of them have fallen into disuse or they are used only in very local environments. For example, only Basque speakers from the Northern side (bilingual Basque and French speakers) are used to pronouncing the Basque “@” (i.e. Sorrapürü). This vowel's pronunciation is between the Basque vocals “u” and “i”. In comparison to Basque or Spanish, French has a very much richer vocal system, but it is fair to say that some of their older forms have fallen into disuse too. Anyway, they keep on being different to those in Basque or Spanish, especially in the case of nasal vowels. In the other hand, some of the consonants that are rare in French such as “L” (i.e. Feuille) are very common in Basque or Spanish. Therefore, a cross-lingual Acoustic Model could be very useful in these cases. Another special feature in this experiment is the richness of affricates and fricatives present in Basque. These sounds will be very difficult to differ and the cross-lingual approach won't work for them, but it has to be said that even native Basque speakers don't make differences between some affricates and fricatives. Consequently, the Acoustic decoder would have difficulties in these cases and further Language Modelling would be needed in order to get accurate results. Finally, some sounds that are differentiated theoretically are very difficult to model, and many state-of-the-art approaches cluster these cases as the same sound. This is the case of the plosives in the three languages; there is little acoustic difference between “b”, “B”, “p”, and “P” depending on the context, and the Language Model should be able to manage the ambiguity derived of not separating those phonemes in this first stage.

Table 1 Sound Inventories for Basque, French and Spanish in the SAMPA notation

Sound Type	Basque	French	Spanish
Plosives	p b t d k g c	p b t d k g	p b t d k g
Affricates	ts ts´ tS		ts
Fricatives	gj jj f B T D s s´ S x G Z v h	f v s z S Z	gj jj F B T D s x G
Nasals	m n J	m n J N	m n J
Liquids	l L r rr	l R	l L r rr
Vowel glides	w j	w H j	w j
Vowels	i e a o u @	i e E a A O o u y 2 9 @ e~ a~ o~ 9~,	i e a o u

3 Description of the Language Resources

The basic resources used in this work have been mainly provided from two Broadcast News sources [7]. On the one hand, the Basque Public Radio-Television group (EITB) has provided us with videos of their Broadcast News in Basque and Spanish. In the other hand, Infozazpi irratia, a new trilingual (Basque, French, and Spanish) digital radio station which currently emits only via Internet has provided audio and text data from their news bulletins.

The speech database described in [8] had to be transcribed orthographically, even when some scripting data existed for Infozazpi irratia files. Thus, the XML label files were created manually. The XML files include information of distinct speakers, noises, and sections of the speech files. The transcriptions for Basque also include morphological information such as each word’s lemma and Part-Of-Speech tag. Resources Inventory is described in Table 2.

Table 2 Resources Inventory: Broadcast News (video, audio), Transcription of broadcast-XML, Text databases and Lexical elements

Language	Broadcast Video	Broadcast Audio	Transcription XML-FILE	Textual Database
EU	6:37	18	12	8M
FR	-	2:58	2:58	2M
ES	9:35	12:34	12:34	4M
Total	16:12	33:12	33:12	14M

4 Experimentation

The first step in the experiments was the extraction of features for the training and test of the models. Features normally used can be grouped in two ways: spectral/energy features, and static/dynamic features. Spectral features are aimed at modelling the spectral envelope, where it is expected to be the maximum phonetic information. Several sets of parameters have been tested taken into account the high level of noise in the signal. Finally a set of 13 Mel Frequency Cepstral and energy and their dynamic components have been selected; the frame period was 10 milliseconds. The FFT uses a Hamming window and the signal had first order pre-emphasis applied using a coefficient of 0.97. The filter-bank had 26 channels and 12 MFCC coefficients (HTK[9]). Each allophonic acoustic unit will be model by a Semi-Continuous Hidden Markov Model (HMM) (HTK).

The first test we performed consisted in the evaluation of the accuracy of the developed APD system. In other words, we tried to determine the allophonic recognition accuracy using just the acoustic models, without any other kind of lexical or grammatical restriction. The second set of tests consisted in trying to determine if it was possible to improve the accuracy selecting the data that threw best results instead of using great amounts of inaccurate data, and the third experiment set combines the three acoustic-phonetic decoders and triphonemes models in order to improve the recognition in Basque using cross-lingual adaptation.

The obtained results in the first set of experiments are presented in Figure 2 (Accuracy) for the three languages. It is obvious that the allophone recognition rate is much better for Spanish and Basque than for French. This can be explained with two main reasons: on the one hand, the simple vowel structure in Basque and Spanish eases the overall recognition of phonemes. On the other hand, the mean noise level in French is also higher and the training data amount is smaller being critical in same cases as affricates.

The second set of test has consisted in ranking the sentences ordered by they error rate in this case with data sets that suit the training task of the decoder better. Specifically, the experiments have been done with the sentences that have a recognition rate over 30%, 40%, 50% and 60% respectively. The sentences that have a recognition rate lower than 30% are in most of the cases noisy or confuse sentences that mislead the training of the system. Paradoxically, when less data is

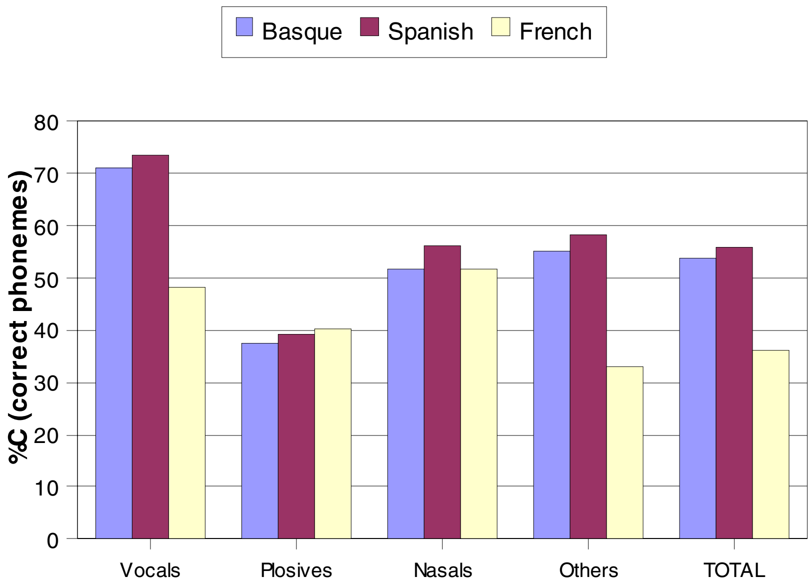


Fig. 2 Results of the experiments clustered by sound types

expected to perform worse because of the Recognition techniques employed, the well selected sentences train better recognizers for many of the allophones, and in some cases they outperform the wider data sets (see figure 3).

Finally, a third set of experiments have been developed trying to apply cross-lingual techniques and triphonemes. The use of triphonemes improves previous results accuracy around %8 and cross-lingual techniques around %3.

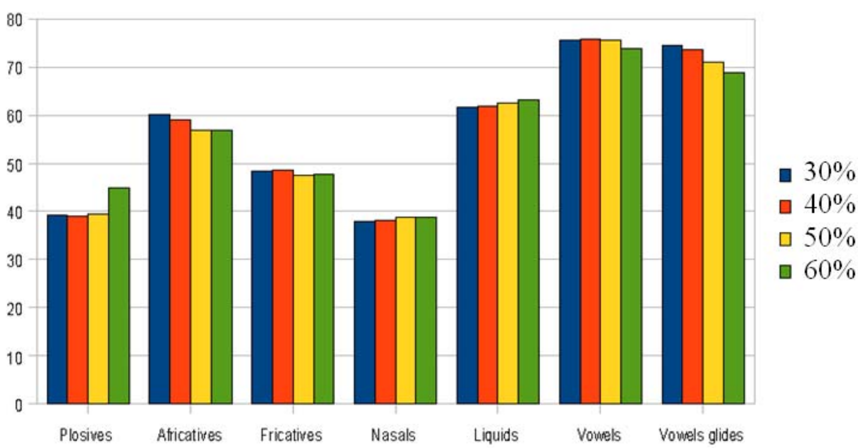


Fig. 3 Accuracy with different data sets

The poor results in some cases (plosives, affricates) could be due to the fact that the covariance matrixes used in the classification are singular (small training set) and, therefore, it turns out impossible to find the inverse matrix. In ongoing works we will use several methods to improve and smooth the covariance matrixes: LOOC estimation method, Bayesian LOOC estimation method (BLOOC) and mix-LOOC estimation method [10,11,12].

5 Concluding Remarks

The final goal of the project is to develop a Multilingual Large Vocabulary Continuous Speech Recognition (MLVCSR) system in Basque context for Spanish, French and Basque. In order to develop a real-life multi-purpose system that could be useful for difficult recognition tasks, we have chosen a Broadcast News Speech Database with many speakers, environments, and noises.

Several Languages Resources have been developed to train and test all the modules of the system. On the one hand, the Basque Public Radio-Television group (EITB) has provided us with videos of their Broadcast News in Basque and Spanish. In the other hand, Infozazpi irratia, a new trilingual (Basque, French, and Spanish) digital radio station which currently emits only via Internet has provided audio and text data from their news bulletins.

A first prototype of Acoustic Phonetic Decoding (APD) system has been developed. This system decodes appropriately the acoustic signal in series of allophonic acoustic units for the three languages: Spanish, French and Basque. Obtained results will be improved in our ongoing work with several methods to calculate and smooth the covariance matrixes when the training sets are small. New cross-lingual methodologies and robustness methods will be also used.

Acknowledgements. This work has been partially supported by the Basque Goverment, SAIOTEK 2007 plan and from the Spanish Ministerio de Ciencia y Tecnología, project TSI2006-14250-C02-01.

References

1. Lopez de Ipiña, K., Graña, M., Ezeiza, N., Hernández, M., Zulueta, E., Ezeiza, A., Tovar, C.: Selection of Lexical Units for Continuous Speech Recognition of Basque. In: Sanfeliu, A., Ruiz-Shulcloper, J. (eds.) CIARP 2003. LNCS, vol. 2905, pp. 244–250. Springer, Heidelberg (2003)
2. Barroso, N., Ezeiza, A., Gilisagasti, N., Lopez de Ipiña, K., López, A., López, J.M.: Development of Multimodal Resources for Multilingual Information Retrieval in the Basque context. In: Proccedings of Interspeech 2007, Antwerp, Belgium (2007)
3. Vandecatseye, A., Martens, J.P., Neto, J., Meinedo, H., Garcia-Mateo, C., Dieguez, F.J., Mihelic, F., Zibert, J., Nouza, J., David, P., Pleva, M., Cizmar, A., Papageorgiou, H., Alexandris, C.: The COST278 pan-European Broadcast News Database. In: Proceedings of LREC 2004, Lisbon, Portugal (2004)

4. Schultz, T., Waibel, A.: Multilingual and Crosslingual Speech Recognition. In: Proceedings of the DARPA Broadcast News Workshop (1998)
5. Wheatley, B., Kondo, K., Anderson, W., Muthusamy, Y.: An Evaluation of Cross-Language Adaptation for Rapid HMM Development in a New Language. In: International Conference on Acoustics, Speech, and Signal Processing, Adelaine, pp. 237–240 (1994)
6. Toledano, D., Moreno, A., Colás, J., Garrido, J.: Acoustic-phonetic decoding of different types of spontaneous speech in Spanish. In: Procceedings of Disfluencies in Spontaneous Speech 2005, Aix-en-Provence, France. (2005)
7. Bordel, G., Ezeiza, A., López de Ipiña, K., Mendez, M., Peñagarikano, M., Rico, T., Tovar, C., Zulueta, E.: Development of Resources for a Bilingual Automatic Index System of Broadcast News in Basque and Spanish. In: LREC 2004, vol. III, pp. 881–884 (2004)
8. Barroso, N., Ezeiza, A., Gilisagasti, N., López de Ipiña, K., López, A., López, J.M.: Development of Multimodal Resources for Multilingual Information Retrieval in the Basque Context. In: Interspeech 2008, Antwerp (2008)
9. Young, S., Jansen, J., Odell, J., Ollason, D., Woodland, P.: The HTK Book. Cambridge U. Press, Cambridge (1997)
10. Hobeck, J.P., Landgrebe, D.: Covariance estimation and classification with limited training data. IEEE Transactions on Pattern Analysis and Machine Intelligence 18(7), 763–767 (1996)
11. Tadjudin, S., Landgrebe, D.: Classification of high dimensional data with limited training samples. Technical Report TR-ECE 98-8. School of Electrical and Computer Engineering, Purdue University, West Lafayette, Indiana (1998)
12. Tadjudin, S., Landgrebe, D.: Covariance Estimation with Limited Training Samples. IEEE Transaction on Geoscience and Remote Sensing 37 (20000029

A Computer-Aided Decision Support System for Shoulder Pain Pathology

K. López de Ipiña, M.C. Hernández, M. Graña, E. Martínez, and C. Vaquero

Abstract. A musculoskeletal disorder is a condition of the musculoskeletal system which consists in that part of it is injured continuously over time. Shoulder disorders are one of the most common musculoskeletal cases attended in primary health care services. Shoulder disorders cause pain and limit the ability to perform many routine activities and affect about 15-25 % of the general population. Several clinical tests have been described to aid diagnosis of shoulder disorders. However, the current literature acknowledges a lack of concordance in clinical assessment, even among musculoskeletal specialists. In this work a Computer-Aided Decision Support (CADS) system for Shoulder Pain Pathology has been developed. The paper presents the medical method and the development of the database and the (CADS) system based on several classical classification paradigms improve by covariance estimation methods. Finally the system was evaluated by a medical specialist.

Keywords: Musculoskeletal disorders, Shoulder Pain, Computer-Aided Decision Support system, Machine Learning.

K. López de Ipiña · M.C. Hernández · M. Graña
Grupo de Inteligencia Computacional
Universidad del País Vasco/Euskal Herriko Unibertsitatea
Plaza de Europa1, Donostia, 20008
e-mail: {karmele.ipina,mamen.hernandez,manuel.graña}@ehu.es

E. Martínez
Txagorritxu Hospital. Rehabilitation Service. Osakidetza, Basque Health Service
C/ José Atxotegi s/n. 01009 Vitoria-Gasteiz
e-mail: elena.martinezgarcia@osakidetza.net

C. Vaquero
Fundación LEIA CDT
Equipo de Seguridad Industrial P.T. Alava – C/ Leonardo Da Vinci, 11, 01510
Miñano, Álava, Spain
e-mail: celinav@leia.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 705–712.
springerlink.com

1 Introduction

A musculoskeletal disorder is a condition of continuous injure over time of a part of the musculoskeletal system. The disorder occurs when the body part is called to work harder, stretch farther, impact more directly or otherwise function at a greater level than it is prepared for. The immediate impact may be minute, but when it occurs repeatedly the accumulated trauma may cause damage [1,2,3]. As reported by recent assessments, especially those made by the Dublin Foundation for the Improvement of Living and Working conditions in its fourth report, Musculoskeletal disorders (MSDs) are the most common work-related health problem in Europe, affecting to millions of workers. Across the EU27, 25% of workers complain of backache and 23% report muscular pains. Although it is not clear to what extent MSDs are caused by work, their impact on working life is huge. MSDs can interfere with activities at work, and can cause a reduction in productivity, an increase in sick leave, and chronic occupational disability. MSDs are caused mainly by manual handling, heavy physical work, awkward and static postures, repetition of movements and vibration. Also, attention should be given to “lack of physical activities” during working time (derived from growing use of visual display units and of automated systems resulting in prolonged sitting at the work place) which has been identified by the European Agency for Safety and Health at Work as an emerging risk. As reported also by this Agency, the risk of MSDs can increase with the pace of work, low job satisfaction, high job demands, job stress and working in cold environments [1,2]. The consequences derived from MSD are quite costly, not only in unmessurable human costs in the form of suffering and functional impairments, but also in economical costs. As reported by the European Agency for Safety and Health at Work, MSDs are the biggest cause of absence from work in practically all Member States. In some states, MSDs account for 40% of the costs of workers’ compensation, and cause a reduction of up to 1.6% in the gross domestic product (GDP) of the country itself. MSDs reduce companies’ profitability and add to the social costs of governments.

Shoulder disorders (SD) are one of the most common primary care musculoskeletal presentation and often recurrence or chronic symptoms may occur in a significant proportion of patients. Shoulder disorders cause pain, limit the ability to perform many routine activities, and can significantly disrupt sleep. Self-reported prevalence of shoulder pain is estimated at around 25% in the general population. Shoulder pain may arise from elsewhere, so it is important to enquire about the general health of the patient and symptoms arising from the neck, upper limbs, axilla and chest.

In the next decade, Computer-Aided Decision Support (CADS) applications for various diagnostic tasks will be developed and then implemented in clinical environments. To date, the majority of CADS applications focus on a limited number of important oncology tasks, mostly related to lung, breast, and colon cancers.

Opportunities exist to expand the scope of CADS applications for organ systems and pathologies previously ignored. For example, new CADS applications for advanced musculoskeletal diagnosis could improve the accuracy and efficiency of human observers, potentially leading to improved patient care. An area of particular interest is orthopaedic imaging [4].

The next section presents the medical method to design the database and the CADS system proposed in this paper. Section 3 describes in detail the database. Section 4 shows results that have been obtained from the classification systems applied to the database and the expert evaluation. Finally, some conclusions and future work are given in section 5.

2 Medical Method

Shoulder Pain (SP) is one of the most frequent pathologies of the locomotors system, being nowadays the third reason of consultation in Primary care, and managing to affect 15-25 % of the adult population. Nevertheless, despite of this impact in the literature it can appreciate a clear disparity of criteria not only in the used terms but also in prediction factors, or in the used treatments. Several clinical tests (e.g. Hawkins, Neer, Yergason, Speed) have been described to aid diagnosis of shoulder disorders. However, research acknowledges a lack of concordance in clinical assessment, even between musculoskeletal specialists [3,5].

In the etiology, it seems clear that the main reasons are mechanical, though diverse factors of risk have considered, as the age, the feminine sex, the physical repetitive work, psychological factors etc. The clinical history is not very effective in the accomplishment of the diagnosis. Regarding to the clinical signs in the exploration the existing ones are numerous, and the evidence shows that those test that has high sensibility, has small specificity, recommending their combination. Inside nowadays medicine trend based on the evidence, several guides of clinical practice for the managing of the shoulder pain have been developed. Among them we can find the developed one for the American Academy of Orthopedic Surgeons, which contemplates six differential diagnoses of the shoulder pain: pathology of the rotator cuff, frozen shoulder, instability glenohumeral, degenerative osteoarthritis glenohumeral, alteration of the joint acromioclavicular and fibromialgia.

Nevertheless the developed one for Duke's University of sports medicine, simplifies this differential diagnosis to four entities: impingement of the rotation cuff, frozen shoulder, instability glenohumeral and degenerative osteoarthritis. Therefore there is no unanimity among the different reasons of shoulder pain. Thus in this work, we have rejected those reasons that are diagnosed well by the causative mechanism, as fractures or luxations, or by the image as the degenerative osteoarthritis. Anyway, they are complex diagnoses and they are overlapped in numerous occasions [3,5,6].

3 Description of the Database

We have selected a sample of 200 patients directed to the Rehabilitation Service for SP. There have been included those patients who after the valuation in consultation have been diagnosed of frozen shoulder, syndrome subacromial (impingement), and rotation cuff pathology. There have been excluded the fractures, the luxations, the above-mentioned pain and the degenerative osteoarthritis glenohumeral, because their diagnosis is mainly radiological.

In the first consultation, we realize the data collection that allows realize a more approximate diagnosis of the pain causes. The collected information is: age, sex, dominance of the arm, characteristics of the pain and its intensity by an analogical visual scale, received treatments, limitation of the mobility and its measure by goniometry, existence of previous history, type of work, sports activity.

In the exploration the mobility has been gathered to articulate by goniometry, the muscular force, and diverse maneuvers for the exploration of the shoulder (empty can test, Hawkins-Kennedy, crossed adducing, apprehension test and subescapular test) [3,5]. These variables have been parameterized to be measured and analyzed in posterior studies. We have gathered some 50 patients diagnosed of syndrome subacromial, some 50 with frozen shoulder and some 50 with diagnosis of rotator cuff pathology.

The database (ACCESS) has been created with SP files from the Txagorritxu's Hospital consists of about 200 files catalogued in 4 different diagnoses: 1) Rotator Cuff Anomaly, 2) SubAcromial Syndrome, 3) frozen Shoulder and 4) Subacromial Syndrome and Rotator Cuff Anomaly. The obtained database is a database balanced for the four classes and labeled and codified in octal system as: 1, 2, 4 and 5. On the other hand files of not common analogous pathologies were also used obtaining 6 types of different diagnoses.

The database has been divided in two independent subsets, the first one for training and the other for test, constituted by 70 not diagnosed cases. The diagnoses qualify in 6 classes. Table 1 shows the number of sample cases for every type of diagnosis.

We have a set of training of about 100 samples, each of which has 30 characteristics. Each of these attributes or characteristics has been identified and the appropriate range has been assigned (Table 2).

Table 1 Number of samples of the 6 classes used for training

	Number of samples
Class1	30
Class2	20
Class3	5
Class4	25
Class5	27
Class6	5

Table 2 Identification and range for the 30 attributes of the database elements

Attribute	Att. Name	Range	Attribute	Att. Name	Range
Attribute0	Edad	[1,2,3]	**Attribute15**	amarotacione	[1,2,3,4]
Attribute1	Sexo	[1,2]	**Attribute16**	amprotacione	[1,2,3,4]
Attribute2	Hombre	[1,2]	**Attribute17**	amarotacioni	[1,2,3,4]
Attribute3	dInicio	[1,2]	**Attribute18**	amprotacioni	[1,2,3,4]
Attribute4	dCronicidad	[1,2]	**Attribute19**	Ritmo	[0,1,2]
Attribute5	dDesencad	[1,2,3]	**Attribute20**	dolorBM	Boolean
Attribute6	Dcantidad	Ŗ	**Attribute21**	Bmaflexion	[1,2,3,4,5]
Attribute7	Dnocturno	Boolean	**Attribute22**	bmaabduccio	[1,2,3,4,5]
Attribute8	tratamientos	[1,2,3]	**Attribute23**	bmarotacione	[1,2,3,4,5]
Attribute9	Movilidad	Boolean	**Attribute24**	bmarotacioni	[1,2,3,4,5]
Attribute10	Previa	Boolean	**Attribute25**	Hawkins	Boolean
Attribute11	Trabajo	[1,2,3,4]	**Attribute26**	Jobe	Boolean
Attribute12	Deporte	Boolean	**Attribute27**	Subescapular	Boolean
Attribute13	ampflexion	[1,2,3]	**Attribute28**	acromicroclav	Boolean
Attribute14	amaflexion	[1,2,3,4]	**Attribute29**	Cervical	boolean

4 Experimentation

We have tested several classification methods: MLE (Maximum Likelihood Estimation), MDM (Minimum Distance to Means), FIS (Fisher's Linear Discriminant), QDF (Quadratic Discriminant Function), KNN (K-Nearest Neighbor), DT (Decision Trees) and two different approaches of SVM (Support Vector Machines), SVM1 and SVM2. The above mentioned methods have been applied over the data training set. In Table 3 it can see the values of the IRATIO index for the selected methods. This value gives the classification accuracy percentage. Results show a good performance for most methods obtaining the best accuracy for SVM2. It must also be highlighted that MLE and QDF obtain very poor results as Table 3 shows.

Table 3 IRATIO validation index and partition number for several training methods

Txagorritxu	Method	Partition
Shoulder Pain	**IRATIO Train**	**Number**
MLE	7,25	2
MDM	46,38	6
FISHER	73,91	6
QDF	7,25	2
KNN	71,01	5
DT	81,16	5
SVM1	95,65	15
SVM2	97,10	15

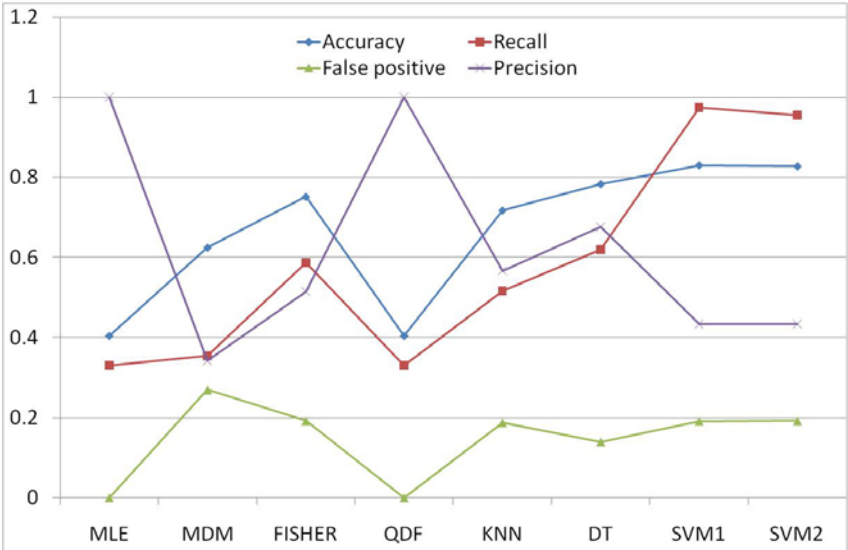


Fig. 1 Analysis of the validation index of the found partitions by the selected methods (Table 3): Accuracy, Recall, False positive and Precision

A study of the validation index over the partitions found by the selected method is detailed below. The analyzed indexes are: Accuracy, Recall, False positive and Precision. Figure 1 confirms the tendency showed in Table 3. The poor results obtained by some of the algorithms comes from the small number of partitions distinguished (Table 3). This is due to the fact that the covariance matrixes used in the classification are singular and, therefore, it turns out impossible to find the inverse matrix. Following we will carry out a new experiment with several me-thods to improve the covariance matrix: MLE, FIS and QDF. The new selected methods are: LOOC estimation method, Bayesian LOOC estimation method (BLOOC) and mix-LOOC estimation method [7,8,9].

Table 4 IRATIO validation index for the covariance matrix estimation methods

Txagorritxu	**Shoulder**	**Pain**	
	MLE IRATIO	**FIS IRATIO**	**QDF IRATIO**
LOOC1	28,99	62,32	27,54
LOOC2	37,68	65,22	37,68
BLOOC1	75,36	52,17	72,46
BLOOC2	76,81	59,42	73,91
mLOOC1	37,68	62,32	37,68
mLOOC2	47,83	57,67	47,83

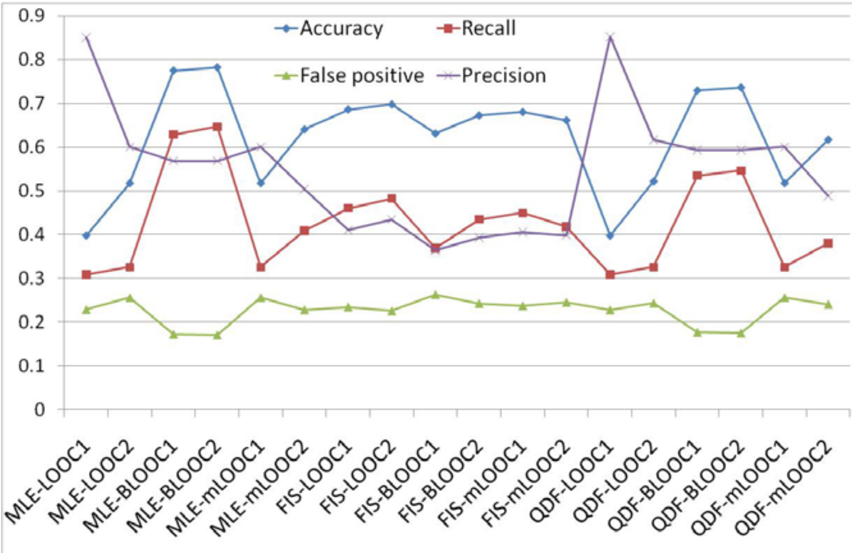


Fig. 2 Analysis of the validation index of the found partitions by the selected methods (Table 4): Accuracy, Recall, False positive and Precision

The above mentioned methods have been applied over the data training set. In Table 4 it can see the values of IRATIO index for the selected methods. New methods improve the rates obtained in Table 3. Figure 2 shows also a best tendency for all validation index.

Finally, a diagnosis evaluation was carried out with the developed system based on the classification optimum method. In this evaluation the independent test (70 cases) was used. The obtained results were supervised by the medical specialist who demonstrates a high degree of satisfaction not only with regard to the visual information provided by the automatic support system but also for the diagnosis results over the test cases. The system has obtained good performance in both supervised and non-supervised modes and results are very close to the diagnosis of the specialist. The specialist prefers in any case the results obtained in supervised mode.

5 Concluding Remarks

Shoulder disorders are one of the most common primary care musculoskeletal presentations and affect to around 25% of the general population. Several clinical tests have been described to aid diagnosis of shoulder disorders. However, research acknowledges a lack of concordance in clinical assessment, even among musculoskeletal specialists. The work presents a Computer-Aided Decision Support (CADS) system for Shoulder Pain Pathology.

The paper deals with the medical method and the development of the database and the (CADS) system based on several classical classification paradigms improve by covariance estimation methods. The system obtains good performance in both supervised and non-supervised modes and results are very close to the diagnosis of the specialist. The specialist prefers in any case the results obtained in supervised mode.

In future outlines, new covariance estimation methods will be developed and the system will be also improved with new non-supervised methods.

Acknowledgements. The involved work has received financial support from the Spanish Ministerio de Ciencia y Tecnología, IBEROEKA 2005 plan and the Spanish Ministerio de Ciencia y Tecnología, project TSI2006-14250-C02-01.

References

1. Gauthy, R.: Muskuloskeletal disorders. An ill-understood "pandemic". ETUI-REHS, Health and Safety Department, Brussels: ETUI-REHS, 2007.- 54 p.; 24 cm, ISBN 978-2-87452-100-3; CDU 616.7
2. European Agency for Safety and Health at Work, Work-related musculoskeletal disorders: Back to work report. Luxembourg, Office for Official Publications of the European Communities (2007)
3. Mitchell, C., Adebajo, A., Hay, E., Carr, A.: Shoulder pain: diagnosis and management in primary care. BM 331(7525), 1124–1128 (2005), `http://www.bmj.com/cgi/content/full/331/7525/1124`
4. Huang, J.-Y., Kao, P.-F., Chen, Y.-S.: A Set of Image Processing Algorithms for Computer-Aided Diagnosis in Nuclear Medicine Whole Body Bone Scan Images. IEEE Transactions on Nuclear Science 54(3), 514–522 (2007)
5. Norlin, R.: Frozen shoulder: etiology, pathogenesis and natural course, October 13 (2005), `http://www.shoulderdoc.co.uk`
6. Ostor, A.J., Richards, C.A., Prevost, A.T., et al.: Diagnosis and relation to general health of shoulder disorders presenting to primary care. Rheumatology (Oxford) 44(6), 800–805 (2005)
7. Hoffbeck, J.P., Landgrebe, D.: Covariance estimation and classification with limited training data. IEEE Transactions on Pattern Analysis and Machine Intelligence 18(7), 763–767 (1996)
8. Tadjudin, S., Landgrebe, D.: Classification of high dimensional data with limited training samples. Technical Report TRECE 98-8. School of Electrical and Computer Engineering, Purdue University, West Lafayette, Indiana (1998)
9. Tadjudin, S., Landgrebe, D.: Covariance Estimation with Limited Training Samples. IEEE Transaction on Geoscience and Remote Sensing 37 (2000)

Unsupervised Visualization of SQL Attacks by Means of the SCMAS Architecture

Álvaro Herrero, Cristian I. Pinzón, Emilio Corchado, and Javier Bajo

Abstract. This paper presents an improvement of the SCMAS architecture aimed at securing SQL-run databases. The main goal of such architecture is the detection and prevention of SQL injection attacks. The improvement consists in the incorporation of unsupervised projection models for the visual inspection of SQL traffic. Through the obtained projections, SQL injection queries can be identified and subsequent actions can be taken. The proposed approach has been tested on a real dataset, and the obtained results are shown.

Keywords: Multiagent System for Security, Neural Projection Models, Unsupervised Learning, Database Security, SQL Injection Attacks.

1 Introduction

Over the last years, one of the most serious security threats to databases has been the SQL injection attack [1]. In spite of being a well-known type of attack, the SQL injection remains at the top of the published threat list [2]. The solutions proposed so far seem insufficient to block this type of attack because the vast majority of them are based on centralized mechanisms [3], [4] with little capacity to work in distributed and dynamic environments. Furthermore, the detection and classification mechanisms proposed by these solutions lack the learning and adaptation capabilities for dealing with attacks and variations of the attacks that may appear in the future.

Álvaro Herrero
Civil Engineering Department, University of Burgos
C/ Francisco de Vitoria s/n, 09006 Burgos, Spain
e-mail: `ahcosio@ubu.es`

Cristian I. Pinzón · Emilio Corchado · Javier Bajo
Departamento de Informática y Automática, Universidad de Salamanca,
Plaza de la Merced s/n 37008, Salamanca, Spain
e-mail : `{cristian_ivanp,jbajope}@usal.es, escorchado@ubu.es`

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 713–720.
springerlink.com

This work presents a novel multiagent solution for anomaly visualization. The proposed multiagent system (MAS) is composed of agents with specialized abilities to detect and predict SQL injection attacks [5]. Most of the agents are focused on data monitoring and analysis. However, it is necessary to incorporate a new agent type with projection ability for anomaly visualization. This agent incorporates different projection models for data visualization, with the aim of notably improving the function of the MAS. As stated in [6], scant attention has been given to visualization in the security field, although visual presentations help operators and security managers to interpret large quantities of data. Several attempts have been made to apply connectionist models to the field of security, mainly based on a classificatory standpoint. A complementary approach is followed in this work, in which the main goal is to provide a data projection to visually identify SQL injection attacks. This idea has been previously applied in the field of Network Intrusion Detection [7].

The rest of the paper is structured as follows: Section 2 introduces the MAS architecture. Section 3 describes the unsupervised projection models. Section 4 shows the experimental results and, finally, Section 5 presents the obtained conclusions and the future work.

2 A Multiagent Solution for SQL Anomaly Visualization

The Structure Query Language (SQL) constitutes the backbone of many Database Management Systems (DBMSs), especially relational databases. It carries out information handling and database management, but it also facilitates building a type of attack that can be extremely lethal. SQL injection attacks are a potential threat at the application layer of the TCP/IP protocol stack. Although this type of attack has been the subject of many studies; it continues to be one of the most frequent attacks over the Internet. SQL injection occurs when the intended effect of the SQL sentence is changed by inserting SQL keywords or special symbols [1].

To deal with such attacks, the SCMAS architecture [5] has been upgraded by including a new type of agent named "Visualizer", which provides the capacity of visualization. Its main function is to complement the classification of SQL attacks through visualization facilities. As a result, this new agent contributes to improving the classification performance of SCMAS. The SCMAS architecture proposes a novel strategy to block SQL injection attacks through a distributed approach based on the capacities of the SQLCBR agents, which are a particular type of CBR-BDI agents [8]. The architecture has been divided into four levels so that the specific tasks are assigned according to the degree of complexity. The different types of agents located at the different levels of the SCMAS architecture can be described as:

- Sensor: captures datagrams, orders TCP fragments to extract the request's SQL string and executes a syntactic analysis of the request's SQL string.

- FingerPrint: performs a pattern matching of known attacks.
- DBPattern: updates and adds new patterns to the pattern database.
- Anomaly: this core component of the architecture carries out a classification of SQL strings through detection anomalies. It integrates a case based reasoning (CBR) mechanism.
- Manager: is responsible for the decision-making, evaluation and coordination of the overall operation of the architecture.
- Forecaster: predicts attacks by considering user behavior.
- LogUser: updates the profile of application users.
- DB: is responsible for executing queries to the database once the requests are classified as legal, and getting the results.
- Interface: facilitates the interaction between a human expert in charge of security and the SCMAS architecture. It can be run on mobile devices.
- Visualizer: this is the new type of agent proposed in this work for upgrading the SCMAS architecture. It applies different projection models for visualizing the SQL-related data, whose features are described in section 4.1. As a consequence of that, the SQL injection attacks can be visually identified.

Fig. 1 depicts the upgraded SCMAS architecture, incorporating the Visualizer agent and showing the different layers and their respective agents.

Fig. 1 Upgraded SCMAS architecture

3 Unsupervised Projection Models

Projection models are used as tools to identify and remove correlations between problem variables, which enable us to carry out dimensionality reduction, visualization or exploratory data analysis. In this study, some unsupervised statistical and neural projection models, namely Principal Component Analysis (PCA) [9], Curvilinear Component Analysis (CCA) [10], and Cooperative Maximum Likelihood Hebbian Learning (CMLHL) [11] have been applied, comparing their results.

PCA [9] is a standard statistical technique for compressing multidimensional data; it can be shown to give the best linear compression of the data in terms of least mean square error. PCA describes the variation in a set of multivariate data in terms of a set of uncorrelated variables each of which is a linear combination of the original variables. Its goal is to derive new variables, in decreasing order of importance, which are linear combinations of the original variables and are uncorrelated with each other.

CCA [10] is a nonlinear dimensionality reduction method. It was developed as an improvement on the Self Organizing Map (SOM) [12], trying to circumvent the limitations inherent in some linear models such as PCA. CCA is performed by a self-organised neural network calculating a vector quantization of the submanifold in the data set (input space) and a nonlinear projection of these quantising vectors toward an output space. As regards its goal, the projection part of CCA is similar to other nonlinear mapping methods, as it minimizes a cost function based on interpoint distances in both input and output spaces. Quantization and nonlinear mapping are separately performed: firstly, the input vectors are forced to become prototypes of the distribution using a vector quantization method, and then, a nonlinear mapping of the input vectors is built.

CMLHL [11] extends the MLHL model [13] that is a neural implementation of Exploratory Projection Pursuit (EPP) [14]. The statistical method of EPP linearly projects a data set onto a set of basis vectors which best reveal the interesting structure in data. CMLHL extends the MLHL model by adding lateral connections [11], which have been derived from the Rectified Gaussian Distribution [15]. Then, CMLHL finds the independent factors of a data set capturing some type of global ordering in the data.

Considering an N-dimensional input vector (x), and an M-dimensional output vector (y), with W_{ij} being the weight (linking input j to output i), then CMLHL can be expressed as:

1. Feed-forward step: $y_i = \sum_{j=1}^{\mathbf{N}} W_{ij} x_j, \forall i$ (1)

2. Lateral activation passing: $y_i(t+1) = \left[y_i(t) + \tau(b - Ay)\right]^+$ (2)

3. Feedback step: $e_j = x_j - \sum_{i=1}^{M} W_{ij} y_i, \forall j$ (3)

4. Weight change: $\Delta W_{ij} = \eta . y_i . sign(e_j) | e_j |^{p-1}$ (4)

Where: η is the learning rate, τ is the "strength" of the lateral connections, b the bias parameter, p a parameter related to the energy function [11], [13] and A is a symmetric matrix used to modify the response to the data [11]. The effect of this matrix is based on the relation between the distances separating the output neurons.

4 Experimental Results

To check the proposed Visualizer agents, several unsupervised projection models (introduced in section 2) have been applied to a real dataset containing samples of the target attacks.

In the projections shown in this section, normal queries are depicted as circles ('O'), while anomalous ones are depicted as crosses ('+'). This "class" information is used only for visualizations purposes. The projection models are not provided with such information while being trained as they are based on unsupervised learning.

4.1 Dataset

A web application with access to a database was developed to check the proposed approach. The web application manages a virtual store where different interfaces are available to carry out queries on the database. MySQL 5.0 was selected as the DBMS to support the web application. Once the database had been created, legal queries were sent from the designed user interfaces. These requests were filtered to avoid redundancy and only legal SQL queries were gathered to generate the dataset. However, in the case of malicious queries, the dispatch of the queries was automated using the tool SQLMap 0.5 [16]. This tool is able to fingerprint an extensive DBMS back-end, retrieve remote DBMS databases, usernames, tables, and columns, enumerate entire DBMS, read system files, and much more taking advantage of web application programming security flaws that lead to SQL injection vulnerabilities. Although the SQLMap 0.5 tool generates a wide variety of malicious queries by using different strategies of attack, these queries were also filtered to remove any similar SQL string previously stored.

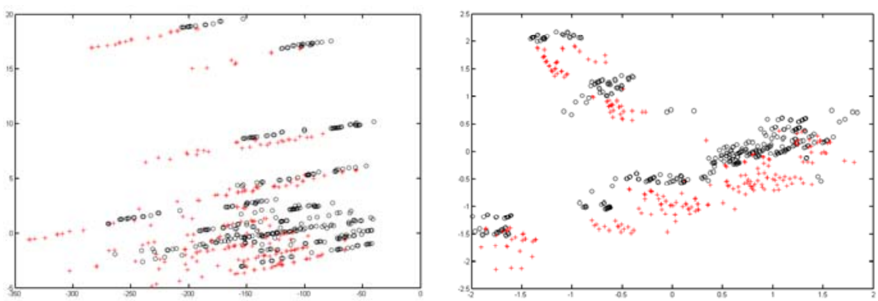
The dataset was formed by a set of 1,000 entries for legal and malicious queries. Finally, for the classification process and application of the projection models, the SQL strings were syntactically analyzed, storing in the dataset the fields described in Table 1.

Table 1. Dataset fields obtained from the syntactic analysis of SQL queries

Field	Description	Type (*Values*)
Affected_table	Number of *tables* affected by the query	Int (*n tables*)
Affected_field	Number of *fields* affected by the query	Int (*n fields*)
Command_type	Type of declared *command* in the query	Int (*0-3*)
Word_GroupBy	Number of repetitions of *Group By* clause	Int (*n clause*)
Word_Having	Number of repetitions of *Having* clause	Int (*n clause*)
Word_OrderBy	Number of repetitions of *Order By* clause	Int (*n clause*)
Numer_And	Number of repetitions of the *And* Operator	Int (*n ops*)
Numer_Or	Number of repetitions of the *Or* Operator	Int (*n ops*)
Number_literals	Number of *Literal* in the SQL string	Int (*n literals*)
Number_LOL	Number of declared Expressions *Literal-Operator-Literal* in the SQL String	Int (*n exprs*)
Length_SQL_String	Length of the SQL String	Int (*n chars*)

4.2 Experiments

The visualization capability of the Analyzer agent was tested by applying its projection models (PCA, CCA and CMLHL) to the aforementioned dataset. Fig. 2 shows the PCA and CMLHL projections of this dataset.



a) PCA projection. b) CMLHL projection.

Fig. 2 Projections of the analysed dataset

The PCA projection (Fig. 2.a) is able to depict some structure of the analyzed dataset. Although these two first principal components amount to 99.7% of data variance, the depicted structure is not related with the normal/anomalous separation of data. That is, normal queries (circles) and anomalous queries (crosses) are mixed up in most of the groups that can be identified in Fig. 2.a.

Fig. 2.b shows the CMLHL projection of the analyzed dataset. Several clusters can be identified in this figure, most of them having a kind of internal organization differentiating normal queries from anomalous ones.

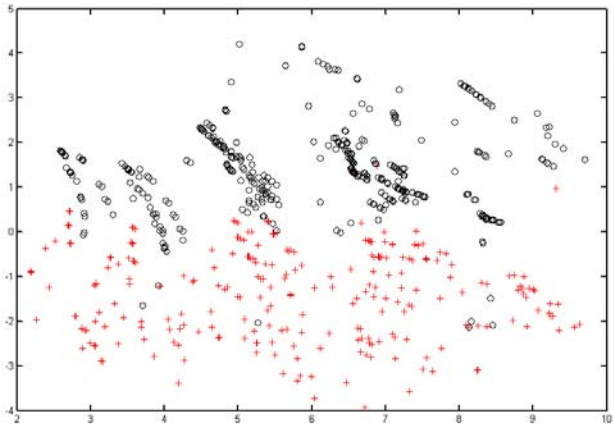


Fig. 3 CCA projection of the analysed dataset

Finally, CCA was applied to the dataset, as can be seen in Fig. 3. This projection model is able to depict the data in a way that most normal queries can be grouped, excluding the anomalous ones. Some normal and anomalous queries are overlapped, what would be considered as false positives/negatives.

From this experimental setup, we can conclude that CCA provides the best projection of the dataset under analysis, outperforming PCA and CMLHL. The CCA projection (Fig. 3) allows the visual identification of anomalous queries, as most of them can be distinguished from normal traffic.

5 Conclusions and Future Work

This paper presents a novel solution based on a new hierarchical MAS for visualizing SQL traffic. It allows the detection of SQL injection attacks by differentiating them from normal SQL queries. This solution combines the advantages of MASs, such as autonomy and distributed problem solving, with the visualization, learning and adaptation capabilities of unsupervised neural projection models. The proposed approach solves one of the lacks of the SCMAS architecture: the visualization of the data in an effective and intuitive way. Further work will focus on the combination of the new visualization abilities with the classification process

Acknowledgments. This research has been partially supported by the Spanish Ministry of Science project TIN2006-14630-C03-03, project of the Junta of Castilla and León BU006A08; Business intelligent for production within the framework of the technological Institute of Castilla y León and investment and services agency (ADE); project of the Spanish Ministry of Education and Innovation CIT-020000-2008-2. The authors would also like to thank the vehicle interior manufacturer, Grupo Antolin Ingenieria S.A., within the framework of the project MAGNO2008 - 1028.- CENIT Project funded by the Spanish Ministry.

References

1. Halfond, W.G.J., Viegas, J., Orso, A.: A Classification of SQL-Injection Attacks and Countermeasures. In: Proceedings of the IEEE International Symposium on Secure Software Engineering, Arlington, VA, USA (2006)
2. Breach Security Inc. The Web Hacking-Incidents Database (2008)
3. Halfond, W.G.J., Orso, A.: AMNESIA: analysis and monitoring for NEutralizing SQL-injection attacks. In: Proceedings of the 20th IEEE/ACM international Conference on Automated software engineering (ASE 2005). ACM, New York (2005)
4. Kosuga, Y., Kono, K., Hanaoka, M., Hishiyama, M., Takahama, Y.: Sania: Syntactic and Semantic Analysis for Automated Testing against SQL Injection. In: 23rd Annual Computer Security Applications Conference. IEEE Computer Society, Los Alamitos (2007)
5. Bajo, J., Corchado, J.M., Pinzón, C., Paz, Y.D., Pérez-Lancho, B.: SCMAS: A Distributed Hierarchical Multi-Agent Architecture for Blocking Attacks to Databases. International Journal of Innovative Computing, Information and Control (2008)
6. Allen, J., Christie, A., Fithen, W., McHugh, J., Pickel, J., Stoner, E.: State of the Practice of Intrusion Detection Technologies. Technical Report CMU/SEI-99-TR-028. Carnegie Mellon University - Software Engineering Institute (2000)
7. Herrero, Á., Corchado, E., Gastaldo, P., Zunino, R.: Neural Projection Techniques for the Visual Inspection of Network Traffic. Neurocomputing 72(16-18), 3649–3658 (2009)
8. Corchado, J.M., Laza, R.: Constructing deliberative agents with case-based reasoning technology. International Journal of Intelligent Systems 18, 1227–1241 (2003)
9. Pearson, K.: On Lines and Planes of Closest Fit to Systems of Points in Space. Philosophical Magazine 2(6), 559–572 (1901)
10. Demartines, P., Herault, J.: Curvilinear Component Analysis: A Self-Organizing Neural Network for Nonlinear Mapping of Data Sets. IEEE Transactions on Neural Networks 8(1), 148–154 (1997)
11. Corchado, E., Fyfe, C.: Connectionist Techniques for the Identification and Suppression of Interfering Underlying Factors. International Journal of Pattern Recognition and Artificial Intelligence 17(8), 1447–1466 (2003)
12. Kohonen, T.: The Self-Organizing Map. IEEE 78(9), 1464–1480 (1990)
13. Corchado, E., MacDonald, D., Fyfe, C.: Maximum and Minimum Likelihood Hebbian Learning for Exploratory Projection Pursuit. Data Mining and Knowledge Discovery 8(3), 203–225 (2004)
14. Friedman, J.H., Tukey, J.W.: A Projection Pursuit Algorithm for Exploratory Data-Analysis. IEEE Transactions on Computers 23(9), 881–890 (1974)
15. Seung, H.S., Socci, N.D., Lee, D.: The Rectified Gaussian Distribution. Advances in Neural Information Processing Systems 10, 350–356 (1998)
16. Damele, B.: SQLMAP0.5 – Automated SQL Injection Tool (2007)

Assessing Knowledge Management in the Power Sector through a Connectionist Model

Álvaro Herrero, Lourdes Sáiz, and Emilio Corchado

Abstract. It has been proven that Artificial Intelligence, in general, and Artificial Neural Networks, in particular, can be successfully applied to problems in the field of Knowledge Management (KM). One such problem is the identification and assessment of a company's KM status. Nowadays the importance of KM to organisational survival and for the maintenance of competitive strength is widely acknowledged. Several connectionist models for the assessment and analysis of KM status are proposed and applied in this work. These models account for the specific features of a company in the Energy sector/Power sector: a dynamic, essential service and one of the basic pillars that supports the so-called "welfare state", constituting an established strategic sector in any globalized economy.

Keywords: Knowledge Management, Exploratory Projection Pursuit, Maximum Likelihood Hebbian Learning, Energy/Power Sector.

1 Introduction

This study of Knowledge Management (KM) centres its attention on the design, organisation and application of innovative and creative ways of resource management, and in particular, those of a more intangible nature [1], [2] until recently left out of traditional business management models. It contributes value to the firm, insofar as that refers to "knowing how to use" resources, rather than merely possessing them, which can achieve satisfactory management of the stock of knowledge that is already held or that may be developed by a firm.

Álvaro Herrero · Lourdes Sáiz
Civil Engineering Department, University of Burgos
C/ Francisco de Vitoria s/n, 09006 Burgos, Spain
e-mail: {ahcosio,lsaiz}@ubu.es

Emilio Corchado
2Departamento de Informática y Automática, University of Salamanca,
Plaza de la Merced s/n, 37008 Salamanca, Spain
e-mail: escorchado@ubu.es

Y. Demazeau et al. (Eds.): Trends in PAAMS, AISC 71, pp. 721–729.
springerlink.com

The application of KM involves the deployment of a set of procedures that allow a firm, on a case-by-case basis, to acquire, capitalize, retain, transfer, and use its strategic knowledge [3]. In essence, the purpose of KM is to identify, acquire, transform, develop, disseminate, use and preserve knowledge that is relevant to achieve specific short and long-term objectives, preferably for the development of new opportunities [4].

On this point, we should set out the differences in terms of "datum", "information " and "knowledge". Data are located in the world, and knowledge is held by the agents (people/organisations). Information is the nexus between both. Knowledge that is acquired, broadened, reviewed, contrasted and used by people in the firm, throughout their working life, is of the greatest value to the organization. Knowledge not only represents the cause, but the result, the outcome and the success of the business activity. In addition, its wealth lies in the variety of knowledge that the firm can amass, apply and develop in its day-to-day activities, along with the social or collective nature of the knowledge, given that a firm's knowledge is not merely a juxtaposition of the individual knowledge of its employees, but a construct that emerges from the events experienced by the firm and its working methods, organizational routines and common values, developed over time.

The identification of knowledge in the form of patterns that exist across dimensional boundaries in high dimensional datasets is a challenging task. Such patterns may become visible if changes are made to the spatial coordinates. Projection models perform such changes by projecting high-dimensional data onto a lower dimensional space in order to identify "interesting" directions in terms of a specific index or projection. These indexes or projections are, for example, based on the identification of directions that account for the largest variance of a dataset – which is the case of Principal Component Analysis (PCA) [5], [6] or the higher order statistics such as the skew or kurtosis index -which is the case of Exploratory Projection Pursuit (EPP) [7]. Having identified the most interesting projections, the data is then projected onto two or three dimensions, which allows its structure to be examined with the naked eye. In this study, several projection models are applied as a KM tool to the Energy/Power Market.

The rest of this paper is organized as follows. Section 2 overviews KM in the Energy/Power sector, while Section 3 describes the main connectionist model applied in this work. Section 4 presents the experimental domain and the results from different unsupervised models. Finally, section 5 arrives at general conclusions and describes future work.

2 Knowledge Management in the Energy/Power Market

The energy sector in the Spanish economy has quite specific features that more than justify the design and application of management models based on the knowledge of the people that constitute the firms in this industry.

Among its special characteristics, the sector's close relationship with a country's industrialization process may be highlighted, its strategic value, with strong "trickle-down effects" for other industries and economic activities, and its tendency to behave according to non-competitive market models. Moreover, energy is implicit in one of the basic pillars that supports the so-called "welfare state", constituting an established strategic sector in any globalized economy.

Owing to the high degree of energy dependence observed in Spain and in Spanish firms, and the broad typology of problems that arise, the efficient application of KM systems is of even greater interest, in order to achieve the correct generation of knowledge, and its transmission, use and adaption to change, which will ensure a satisfactory and reliable supply of electricity. At the same time, the energy sector holds associated knowledge, values that underpin its organizational culture, part of its know-how and technical and organizational capacity, which are unique and are not found in other sectors. Along with this, a great part of the knowledge and ability needed to develop the work in an effective and, above all, in a safe way can only be acquired within the sector. A great amount of this knowledge and these abilities need years of training and experience.

The introduction of a KM model in energy firms [8], [9], will enable key, sector-specific knowledge and abilities to be identified. It will facilitate their correct generation and complete application, the inventory of expert people, the retention and capitalization of experience and “know how”, beyond that held by the expert, sharing and transfer between the people involved and the need for training and updating, arising from new needs that appear every day, such as regulatory requirements, improvements in technical capacities or greater operative experience.

3 Connectionist Projection Model

The solution proposed in this paper applies an unsupervised neural model called Cooperative Maximum Likelihood Hebbian Learning (CMLHL) [10], [11]. It is based on Maximum Likelihood Hebbian Learning (MLHL) [10], [12]. Considering an N-dimensional input vector (x), and an M-dimensional output vector (y), with W_{ij} being the weight (linking input j to output i), then CMLHL can be expressed as:

1. Feed-forward step: $y_i = \sum_{j=1}^{N} W_{ij} x_j, \forall i$ (1)

2. Lateral activation passing: $y_i(t+1) = [y_i(t) + \tau(b - Ay)]^+$ (2)

3. Feedback step: $e_j = x_j - \sum_{i=1}^{M} W_{ij} y_i, \forall j$ (3)

4. Weight change: $\Delta W_{ij} = \eta . y_i . sign(e_j) | e_j |^{p-1}$ (4)

Where: η is the learning rate, τ is the “strength” of the lateral connections, b the bias parameter, p a parameter related to the energy function [10], [11], [12] and A a symmetric matrix used to modify the response to the data [11]. The effect of this matrix is based on the relation between the distances separating the output neurons.

4 Case Study

The empirical study presented in this work is based on the analysis of a set of data related to the introduction of KM in an energy sector firm, using projection methods. The data sample totals 229 representative records (samples) on the status of critical knowledge and its management in the firm under study.

Each record covers information on the task or function assigned to each employee and the knowledge that is needed to carry out that task correctly. The tasks are analyzed according to their degree of difficulty, their impact on the business, who receives their results and the way in which they are received. In contrast, the characteristics of the knowledge under study are difficulty of transfer, its importance to the work, whether it is documented, the knowledge level of the expert, the degree to which it is known and shared with other employees, the method used to transfer it and the urgency of its acquisition, in case of a shortfall or a lower level than that required being reached. It should be pointed out that because of the type of firm under study, the parameters on the degree of documentation and the knowledge level held were not discriminatory, which is to say, the necessary key knowledge is held within the firm and is documented.

In this study, the objective of the analysis is to cross-reference the data against common characteristics or parameters, from both the tasks and their associated knowledge, in order to construct a representative map of tasks and the key knowledge and their immediate visualization, according to their situation on the map. All of that will serve to take more accurate decisions when setting up a KM system in harmony with the firm's selected strategy.

4.1 Description of the Problem under Study

The data sample under study amounts to 229 records that correspond to a total of 120 tasks/functions and 142 critical pieces of knowledge gathered from among the employees of a firm. This dataset represents the knowledge state of the firm and, once the data is processed and the corresponding conclusions drawn, these lead to proposals for actions that need to be taken in relation to retention, protection and capitalization of key knowledge for the competitiveness of the business.

For each of the tasks, the knowledge state comprises the aforementioned information referring to the degree of difficulty associated with its execution, the set of knowledge that is applied, its impact on key aspects of the firm, who will receive the result of carrying out each task and how it will be received. This set of data is completed with the detail and the characteristics presented by the knowledge that is assigned to each task and the data relating to the receptor of the knowledge. Thus, the following features are gathered for each piece of knowledge:

- Level of knowledge acquired, ranging from 1 (not enough) to 4 (expert).
- Difficulty of the knowledge, ranging from 1 (lowest) to 5 (highest).
- Receptors of the knowledge, generating 4 binary features (section/department/company/external partners).

- Documented or otherwise, binary values: 1 (documented) and 0 (undocumented).
- Importance to the task, ranging from 1 (medium) to 5 (critical).
- Urgency of knowledge transfer, ranging from 1 (6 months) to 5 (immediate).
- Difficulty of knowledge transfer, ranging from 1 (low) to 5 (high).
- Method of transfer, generating 6 binary features (Written/Oral/On-the-job observation/Training/Inactive/Practicing).
- Number of people that share the knowledge.
- Receptor required training level, ranging from 0 (none) to 2 (university degree).
- Consequence of knowledge absence, ranging from 1 (lowest) to 5 (highest).

As a result, 24 features for each one of the records are included in the dataset.

4.2 Experiments and Results

CMLHL was applied in order to analyse the dataset described above and to identify its inner structure. The best CMLHL projection is shown in Fig. 1.

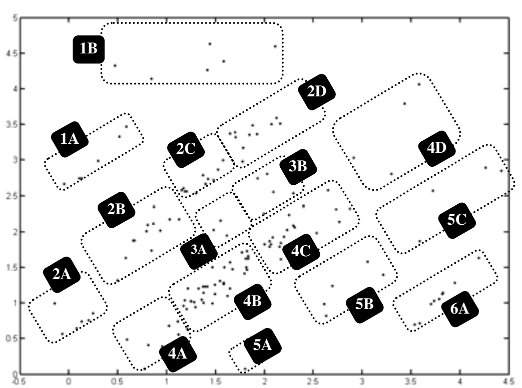


Fig. 1 CMLHL projection

The following interesting conclusions may be drawn from the position of the points (knowledge-tasks) on the projection. Starting with the "Importance of knowledge to the task", of the three possible options -indispensable, important and average- it may be seen that the points that are found in the upper third of the projection come under the heading of "indispensable", the central positions represent important knowledge and those in the lower third are of average importance. The results show that the largest part of the identified knowledge is either important or is of average importance.

With regard to "Knowledge sharing" within the firm, it shows records that go from less to more; less dispersed knowledge is situated in the upper part of the projection and the most widely shared knowledge, in the lower part. Linking this latter result with the former one, it may be concluded that the least important knowledge is the most widely shared, the contrary being true in the case of indispensable knowledge.

The variable “Difficulty of knowledge transfer” covers three possible values that are high, medium and low. High difficulty is represented by the points in the upper third of the projection, the central area covers knowledge of medium difficulty and the points in the lower third represent knowledge of the lowest difficulty. It may therefore be seen that most of the knowledge under analysis has a transfer difficulty of between medium and low, as well as presenting intermediate levels of importance and a high degree of sharing.

The “Consequence of not holding the knowledge” provides information on the effects and their impact on the tasks due to a lack of necessary knowledge; on a barometer of serious, moderate and minor. Here, the distribution of the points reveals that the knowledge situated to the mid-left of the projection presents serious consequences, a parameter which changes as we move to the right.

With regard to the characteristic that refers to “Urgency for knowledge transfer”, three possible responses were foreseen that are ‘immediate’, ‘six-months’ and ‘over six-months’. CMLHL allows us to observe a vertical trend by which the upper positions reflect an immediate urgency, which moving down to the centre drops to 6 months, and continues to drop down to lower levels, reaching the lowest value, which is over 6 months. Examining the conclusions reached up until this point, this last point on the urgency of transfer shows that critical knowledge is the least shared, presents the greatest difficulty in its transfer and, in addition, is revealed to be of immediate urgency. For its part, knowledge of average importance neither presents as much difficulty, nor is such a short time required for its transfer and, furthermore, the analysis revealed that it is more evenly shared than the other types of knowledge.

With regard to the “Method of knowledge transfer”, the positions of the points allow us to conclude very clearly that those on the left represent situations centred on “training” and “on-the-job observation”; knowledge that is more efficiently shared through “practicing” is positioned in the centre and the “written” and “oral” methods are positioned to the right.

Integrating the earlier results on the areas in which the knowledge may be grouped (1A, 1B, 2A, 2B, 2C, 2D, 3A, 3B, 4A, 4B, 4C, 4D, 5A, 5B, 6A), according to its position (Fig. 1), the earlier analysis may be complemented by looking at the features and consequences presented by the knowledge according to the position in each of the aforementioned sets.

The knowledge in group 1A is indispensable, there is a slight tendency to sharing, whereas the consequences of its absence are high, the difficulty of transfer is equally high and the urgency of the knowledge that is not held is almost immediate. This set of data represents a weak situation with regard to sharing and diffusion of knowledge within the firm.

Something very similar may be said about the points that make up group 1B, except that its difficulty of transfer is not so clearly appreciated and sharing is still less than in group 1A. Equally, such results represent situations in which knowledge is lacking, and they hold negative consequences for the firm in the case of a lack of knowledge.

Knowledge of average importance is found around the fringes of 2A, which is reasonably well shared, its difficulty of transfer is minor, as is the case for the urgency of its transfer and the consequences of a lack of knowledge are situated between serious and moderate. All the parameters, except the importance of knowledge, reveal a situation that is close to optimal. As shown in the comparison between groups 1A and 1B, in this case, the situations that 2B and 2C represent are almost identical to those described in 2A; in both the exception is given by the degree of importance attributed to knowledge that is a little greater, and the same may be said for the difficulty of its transfer.

Group 2D brings together the strategic knowledge for the tasks, which is not however shared between the employees; its difficulty being high and its urgency of transfer serious. This group reveals the need to adopt the necessary means to capitalize and share key knowledge, either through training, practicing with the expert, overlapping activities with the others or temporarily changing activities.

Groups 3A and 3B coincide in that they both have indispensable and important knowledge, although these are more evenly shared in 3A, and likewise the difficulty is lower and its urgency of transfer is minor. The consequences of a lack of knowledge are presented at almost the same level, which is medium.

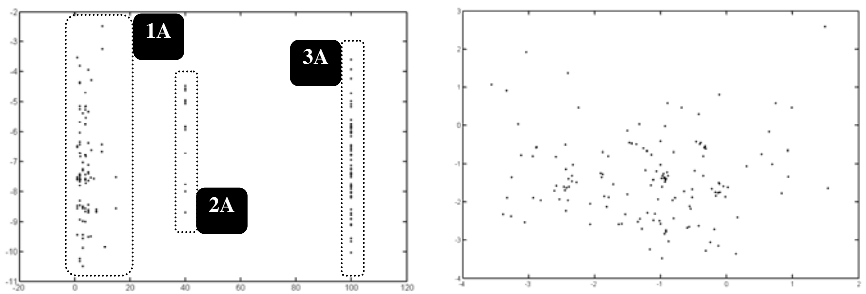
Less importance is given to knowledge in Group 4A: it is shared, the difficulty of transfer is low and the urgency for the knowledge is not immediate. Thus, the study reveals that the consequences of a lack of knowledge, where applicable, reach an intermediate value. The records found in groups 4B and 4C, both represent very similar situations: 4C has a slightly better situation with respect to the importance of knowledge for the task, but a worse one with respect to lower values for sharing knowledge and better results under the heading of difficulty of transfer. The group 4D, has notably better results for the quality of knowledge, but worse results for its sharing and difficulty of transfer with respect to 4C.

Group 5A contains a single point that can be considered an outlier, while group 5B brings together knowledge of medium importance for the tasks that is fully shared, and that has a transfer difficulty of almost zero and the lowest possible urgency. The consequences of its absence are very minor. It is verging on an optimal situation. In contrast, knowledge is more important in 5C, although it is also shared and the difficulty of its transfer is a little greater.

Finally, group 6A brings together the most important but not indispensable knowledge, which is largely shared within the firm, whereas its difficulty and urgency of transfer reach lower values and the consequences of its absence are minor.

4.3 Comparative Study

For comparative purposes, some other projection techniques, namely PCA (see section 4.3.1) and MLHL (see section 4.3.2), were applied to the previously described dataset. The obtained projections are shown in Fig. 2.



a) PCA projection. a) MLHL projection.

Fig. 2 Projections of the analysed dataset

As can be seen in Fig. 2.a, the PCA [5] [6] projection is able to identify a general structure of the dataset consisting of 3 main groups (1A, 2A, and 3A). After analysing the samples in each one of these groups, it can be concluded that only the feature "Number of people that share the knowledge" in the original dataset was taken into account when projecting that structure. This feature provides the greatest variance in the dataset and, as a result, PCA has identified it as the first principal component.

As can be seen in Fig. 2.b, the MLHL projection is not as good as the CMLHL projection (Fig. 1), as it shows the inner structure of the dataset less clearly. MLHL provides a sparse projection of the data but sharply defined groups can not be identified as in the case of CMLHL.

5 Conclusions and Future Work

An assessment of KM is especially critical in the Energy/Power sector. This work has proposed a tool based on a connectionist model for the assessment and analysis of the KM status in a company which takes the specific features of the Energy/Power sector into account. As can be seen in section 4.2, the CMLHL model enabled KM managers to describe the KM status of a power company. This can be used to analyse and consequently improve the KM process of the whole company. It has been also shown that CMLHL outperforms PCA and MLHL in obtaining the most useful projection of a given dataset for KM purposes.

Future work will focus on the upgrade of the proposed model to automate the proposals to improve the KM status by applying different techniques in a Hybrid Artificial Intelligence System.

Acknowledgments. This research is partially supported through the *Junta de Castilla and León* project BU006A08, Business intelligence for production within the framework of the *Instituto Tecnológico de Castilla y León* (ITCL) and the *Agencia de Desarrollo Empresarial* (ADE), and the Spanish Ministry of Education and Innovation project CIT-020000-2008-2. The authors would also like to thank the vehicle interior manufacturer, Grupo Antolin Ingenieria S.A., within the framework of the project MAGNO2008 - 1028.-CENIT Project funded by the Spanish Government.

References

1. Collis, D.J., Montgomery, C.A.: Competing on Resources: Strategy in the 1990s. Harv. Bus. Rev. 73(4), 118–128 (1995)
2. Maranto-Vargas, D., Rangel, R.G.T.: Development of Internal Resources and Capabilities as Sources of Differentiation of SME under Increased Global Competition: A Field Study in Mexico. Technol. Forecast. Soc. Chang. 74(1), 90–99 (2007)
3. Wiig, K.M.: Knowledge Management Foundations: Thinking about Thinking-how People and Organizations Represent, Create, and Use Knowledge. Schema Press, Limited (1994)
4. Soo, C.W., Devinney, T.M., Midgley, D.F.: External Knowledge Acquisition, Creativity and Learning in Organisational Problem Solving. International Journal of Technology Management 38(1), 137–159 (2007)
5. Pearson, K.: On Lines and Planes of Closest Fit to Systems of Points in Space. Philosophical Magazine 2(6), 559–572 (1901)
6. Oja, E.: Neural Networks, Principal Components, and Subspaces. International Journal of Neural Systems 1, 61–68 (1989)
7. Friedman, J.H., Tukey, J.W.: A Projection Pursuit Algorithm for Exploratory Data-Analysis. IEEE Transactions on Computers 23(9), 881–890 (1974)
8. Metaxiotis, K.: Better Management of Energy Knowledge: the Key for Success in the Energy Sector. International Journal of Computer Applications in Technology 22(2/3), 104–108 (2005)
9. Carpenter, S., Rudge, S.: Benchmarking KM at British Energy. Knowledge Management 6(6), 3 (2003)
10. Corchado, E., MacDonald, D., Fyfe, C.: Maximum and Minimum Likelihood Hebbian Learning for Exploratory Projection Pursuit. Data Mining and Knowledge Discovery 8(3), 203–225 (2004)
11. Corchado, E., Fyfe, C.: Connectionist Techniques for the Identification and Suppression of Interfering Underlying Factors. International Journal of Pattern Recognition and Artificial Intelligence 17(8), 1447–1466 (2003)
12. Fyfe, C., Corchado, E.: Maximum Likelihood Hebbian Rules. In: 10th European Symposium on Artificial Neural Networks, ESANN 2002 (2002)

Author Index