ADVANCES IN INTELLIGENT AND SOFT COMPUTING 156

Javier Bajo Pérez et al. (Eds.)

Highlights on Practical Applications of Agents and Multi-Agent Systems

Springer

Advances in Intelligent and Soft Computing 156

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

For further volumes:
http://www.springer.com/series/4240

Javier Bajo Pérez, Juan M. Corchado Rodríguez,
Emmanuel Adam, Alfonso Ortega, María N. Moreno,
Elena Navarro, Benjamin Hirsch,
Henrique Lopes-Cardoso, Vicente Julián,
Miguel A. Sánchez, and Philippe Mathieu (Eds.)

Highlights on Practical Applications of Agents and Multi-Agent Systems

10th International Conference on Practical Applications of Agents and Multi-Agent Systems

Springer

Editors
Javier Bajo Pérez
Facultad de Informática
Universidad Pontificia de Salamanca
Salamanca, Spain

Juan M. Corchado Rodríguez
Departamento de Informática y Automática
Facultad de Ciencias
Universidad de Salamanca
Salamanca, Spain

Emmanuel Adam
LAMIH (UMR CNRS 8530)
Universite de Valenciennes, France

Alfonso Ortega
Departamento de Ingeniería Informática
Calle Francisco Tomás y Valiente
Ciudad Universitaria de Cantoblanco
Madrid, Spain

María N. Moreno
Departamento de Informática y Automática
Facultad de Ciencias
Universidad de Salamanca
Salamanca, Spain

Elena Navarro
Departamento de Informática
University of Castilla-La Mancha
Albacete, Spain

Benjamin Hirsch
EBTIC / Khalifa University
Abu Dhabi, United Arab Emirates

Henrique Lopes-Cardoso
Departamento de Engenharia Informática
Faculdade de Engenharia
Universidade do Porto, Porto, Portugal

Vicente Julián
Departamento de Sistemas Informáticos
y Computación
Universidad Politécnica de Valencia
Valencia, Spain

Miguel A. Sánchez
Facultad de Informática
Universidad Pontificia de Salamanca
Salamanca, Spain

Philippe Mathieu
Université des Sciences et Technologies
de Lille, France

ISSN 1867-5662 e-ISSN 1867-5670
ISBN 978-3-642-28761-9 e-ISBN 978-3-642-28762-6
DOI 10.1007/978-3-642-28762-6
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2012933089

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

PAAMS'12 Special Sessions 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 that emphasized on multi-disciplinary and transversal aspects, as well as cutting-edge topics were especially encouraged and welcome.

Research on Agents and Multi-Agent Systems has matured during the last decade and many effective applications of this technology are now deployed. An international forum to present and discuss the latest scientific developments and their effective applications, 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 evolution of the International Workshop on Practical Applications of Agents and Multi-Agent Systems. PAAMS is an international yearly tribune 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 2012 edition in the special sessions: COREMAS: COoperative and RE-configurable MultiAgent System for Manufacturing, Logistics and Services domains in Industry, AMMR: Adaptive Multimedia and Multilingual Retrieval, BioMAS: 3rd International Special Session on Bio-Inspired and Multi-Agents Systems: Applications to Languages (BioMAS), WebMiRes: Web Mining and Recommender systems, MASSS: Multi-Agent Systems for Safety and Security, AAA: Assessing Agent Applications), TINMAS: Trust, Incentives and Norms in open Multi-Agent Systems, AMAS: Adaptive Multi-Agent Systems, ASM: Agents for smart mobility, ABAM: Agents Behaviours and Artificial Markets.

We would like to thank all the contributing authors, as well as the members of the Program Committees of the Special Sessions and the Organizing Committee for their hard and highly valuable work. Their work has helped to contribute to the success of the PAAMS'12 event. Thanks for your help, PAAMS'12 wouldn't exist without your contribution.

We thank the sponsors (IEEE Systems Man and Cybernetics Society Spain, AEPIA Asociación Española para la Inteligencia Artificial, APPIA Associação Portuguesa Para a Inteligência Artificial, CNRS Centre national de la recherche scientifique), the Local Organization members and the Program Committee members for their hard work, which was essential for the success of PAAMS'12.

Juan M. Corchado Rodríguez
Javier Bajo
PAAMS'12 Organizing Co-chairs

Organization

Special Sessions

COREMAS: COoperative and RE-configurable MultiAgent System for Manufacturing, Logistics and Services domains in Industry.
BioMAS: 3rd International Special Session on Bio-Inspired and Multi-Agents Systems: Applications to Languages.
WebMiRes: Web Mining and Recommender systems.
MASSS: Multi-Agent Systems for Safety and Security.
AAA: Assessing Agent Applications.
TINMAS: Trust, Incentives and Norms in open Multi-Agent Systems.
AMAS: Adaptive Multi-Agent Systems.
ASM: Agents for Smart Mobility.
ABAM: Agents Behaviours and Artificial Markets.

Special Session on COoperative and RE-configurable Multi-Agent System for Manufacturing, Logistics and Services Domains in Industry

Emmanuel Adam	University of Valenciennes, France
Paulo Leitao	Polytechnic Institute of Bragança, Portugal

Special Session on 3rd International Special Session on Bio-Inspired and Multi-Agents Systems: Applications to Languages

Gemma Bel-Enguix	Universitat Rovira i Virgili, Spain
Mª Dolores Jiménez López	Universitat Rovira i Virgili, Spain
Alfonso Ortega de la Puente	Universidad Autónoma de Madrid, Spain

Special Session on Web Mining and Recommender Systems

María N. Moreno (Chair)	University of Salamanca, Spain
Valérie Camps (Chair)	University of Toulouse, France
Ana María Almeida	Institute of Engineering of Porto, Portugal
Anne Laurent	University of Montpellier 2, France
Chris Cornelis	Ghent University, Belgium
Constantino Martins	Institute of Engineering of Porto, Portugal
Maguelonne Teisseire	University of Montpellier 2, France
María José del Jesús	University of Jaen, Spain
Marie-Aude Aufaure	Ecole Centrale de Paris, France
Rafael Corchuelo	University of Sevilla, Spain
Vivian López Batista	University of Salamanca, Spain
Joel Pinho Lucas	HP, Brazil
Salima Hassas	University of Lyon, France
Sylvain Giroux	University of Sherbrooke, Canada
Yolanda Blanco	University of Vigo, Spain
Jérôme Picault	Alcatel-Lucent Bell Labs, France

Special Session on Multi-Agent Systems for Safety and Security

Antonio Fernández-Caballero (Chair)	University of Castilla-La Mancha, Spain
Elena María Navarro Martínez (Chair)	University of Castilla-La Mancha, Spain
Ruth Aguilar-Ponce	Universidad Autónoma de San Luis Potosí, Mexico
Javier Albusac	Universidad de Castilla-La Mancha, Spain
Giuliano Armano	University of Cagliari, Italy
Danilo Avola	Sapienza University of Rome, Italy
Federico Castanedo	Universidad de Deusto, Spain
Rita Cucchiara	Università degli Studi di Modena e Reggio Emilia, Italy
Francisco Garijo	Institut de Recherche en Informatique de Toulouse, France
Marie-Pierre Gleizes	Institut de Recherche en Informatique de Toulouse, France
Zahia Guessoum	Laboratoire d'Informatique de Paris 6, France
Alaa Khamis	The German University in Cairo, Egypt
José Manuel Molina	Universidad Carlos III de Madrid, Spain
Takashi Matsuyama	Kyoto University, Japan
Paulo Novais	Universidade do Minho, Portugal
Juan Pavón	Universidad Complutense de Madrid, Spain
Milind Tambe	University of Southern California, USA

Special Session on Assessing Agent Applications

Benjamin Hirsch	EBTIC / Khalifa University, United Arab Emirates
Tina Balke	University of Bayreuth, Germany
Marco Lützenberger	DAI-Labor, Technical University of Berlin, Germany

Special Session on Trust, Incentives and Norms in open Multi-Agent Systems

Ana Paula Rocha (Chair)	University of Porto, Portugal
Henrique Lopes Cardoso (Chair)	University of Porto, Portugal
Olivier Boissier (Chair)	ENSM Saint-Etienne, France
Ramón Hermoso (Chair)	Universidad Rey Juan Carlos, Spain
Daniel Villatoro	IIIA-CSIC, Spain
Eugénio Oliveira	University of Porto, Portugal
Holger Billhardt	Universidad Rey Juan Carlos, Spain
Joana Urbano	University of Porto, Portugal
Jordi Sabater-Mir	IIIA-CSIC, Spain
Laurent Vercouter	University of Rouen, France
Nicoletta Fornara	University of Lugano, Switzerland
Sascha Ossowski	Universidad Rey Juan Carlos, Spain
Tibor Bosse	Vrije Universiteit Amsterdam, The Netherlands
Virginia Dignum	Delft University of Technology, The Netherlands
Wamberto Vasconcelos	University of Aberdeen, The Netherlands
Yao-Hua Tan	Delft University of Technology, The Netherlands

Special Session on Adaptive Multi-Agent Systems

Vicente Julian	Polytechnic University of Valencia, Spain
Alberto Fernández	University of Rey Juan Carlos, Spain
Juan M. Corchado	University of Salamanca, Spain
Vicente Botti	Polytechnic University of Valencia, Spain
Sascha Ossowski	University of Rey Juan Carlos, Spain
Sara Rodríguez González	University of Salamanca, Spain
Carlos Carrascosa	Polytechnic University of Valencia, Spain
Javier Bajo	Pontifical University of Salamanca, Spain
Martí Navarro	Polytechnic University of Valencia, Spain

Stella Heras	Polytechnic University of Valencia, Spain
Carlos A. Iglesias	Technical University of Madrid, Spain
Holger Billhardt	University King Juan Carlos I, Spain
Javier Palanca	Polytechnic University of Valencia, Spain
Juan Francisco de Paz Santana	University of Salamanca, Spain
Álvaro Herrero	University of Burgos, Spain
Victor Sanchez	Polytechnic University of Valencia, Spain

Special Session on Agents for Smart Mobility

Miguel Angel Sánchez Vidales	Universidad Pontificia de Salamanca, Spain

Special Session on Agents Behaviours and Artificial Markets

Philippe Mathieu (Chair)	University Lille, France
Bruno Beaulfis	University of Lille1, France
Olivier Brandouy	University of Paris 1, France
Andrea Consiglio	University of Palermo, Italy
Florian Hauser	University of Innsbruck, Austria
Juan Pavón	University Complutense Madrid, Spain
Marco Raberto	University of Rekjavik, Iceland
Roger Waldeck	Telecom Bretagne, France
Hector Zenil	University of Sheffield, United Kingdom

Organizing Committee

Juan M. Corchado (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
Fernando de la Prieta Pintado	University of Salamanca, Spain
Davinia Carolina Zato Domínguez	University of Salamanca, Spain

Reviewers

Emmanuelle Grislin
Gauthier Picard
Marin Lujak
Pavel Vrba
Carole Bernon
Jose Barata
Andres Garcia
Jean-Paul Jamont
Zahia Guessoum
Valérie Camps
Montserrat Mateos
Encarnación Beato
Roberto Berjón

Contents

COREMAS: COoperative and RE-configurable Multi Agent System for Manufacturing, Logistics and Services Domains in Industry

BioMAS: 3rd International Special Session on Bio-Inspired and Multi-Agents Systems: Applications to Languages (BioMAS)

WebMiRes: Web Mining and Recommender systems

MASSS: Multi-Agent Systems for Safety and Security

AAA: Assessing Agent Applications

TINMAS: Trust, Incentives and Norms in Open Multi-Agent Systems

AMAS: Adaptive Multi-Agent Systems

ASM: Agents for Smart Mobility

ABAM: Agents Behaviours and Artificial Markets

Multilevel MAS Architecture for Vehicles Knowledge Propagating

Emmanuel Adam, René Mandiau, and Emmanuelle Grislin

Abstract. Completely autonomous vehicles in traffic should allow to decrease the number of road accident victims greatly, and should allow gains in terms of performance and economy. Models of the interaction among the different vehicles is one of the main challenges. We propose in this paper a model of communication of knowledge between mobile agents on a traffic network. The model of knowledge and of interaction enables to propagate new knowledge without overloading the system with a too large number of communications. For that, only the new knowledge is communicated, and two agents communicate the same knowledge only once. In order to allow agents to update their knowledge (perceived or created), a notion of degradation is used. A simulator has been built to evaluate the proposal.

1 Introduction

In many studies on traffic supervision, the optimization of traffic flow as well as new road infrastructure [2], attempt to deal with collective interests and individual interests. We think that completely autonomous vehicles in traffic should allow to decrease the number of road accident victims greatly, and should allow gains in terms of performance and economy. Developing models of the interaction among the different vehicles is one of the main challenges [4] to optimize traffic flow with autonomous vehicles.

The simulation of traffic is often used to evaluate traffic flow optimization methods. In the area of road traffic simulation, two approaches allow management

Emmanuel Adam · René Mandiau · Emmanuelle Grislin
Lille Nord de France, F-59000 Lille, France
e-mail: {emmanuel.adam,rene.mandiau}@univ-valenciennes.fr,
emmanuelle.grislin@univ-valenciennes.fr

Emmanuel Adam · René Mandiau · Emmanuelle Grislin
UVHC, LAMIH, F-59313 Valenciennes, France

Emmanuel Adam · René Mandiau · Emmanuelle Grislin
CNRS, UMR 8201, F-59313 Valenciennes, France

J.B. Pérez et al. (Eds.): Highlights on PAAMS, AISC 156, pp. 1–8.
springerlink.com

policies for scheduling vehicles flows: centralized approaches and distributed approaches. A means to bypass the limitations of centralized approaches (lack of flexibility, difficulty to adapt to changes ...) is to decentralize the traffic simulation [9, 7]; in this context agent-based approaches seem to be the most appropriate [5, 6]. In these approaches, the traffic is the result of the sum of all actions and interactions of the various simulated entities (busses, vehicles, pedestrians, road signs, road infrastructure for examples).

So, in the context of a project that aims at studying the impact of information communication between drivers and infrastructure or elements of the environment (like shops, car park for instance), we turn to multiagent systems to propose a simulator of a traffic network.

Communication of knowledge implies classically to take care of the confidence about this knowledge. If β_a is the set of knowledge of the agent a. A received knowledge b can be :

- out-of-date : dynamic environment implies to date the knowledge about it, about events.
- from a doubtful origin : the knowledge comes from an unknown agent, or without the required signature.
- inconsistent : the receiver of the knowledge b is is unable to store it in its own knowledge without getting an inconsistency; $(\{b\} \cup \beta_a = \emptyset)$.

The next section of the paper presents the architecture of the multiagent system used to share knowledge between mobile agents and the model of knowledge that we propose. The section 3 presents the simulator of road traffic, and the case study that we used to validate our proposal. The last section draws our conclusions and gives some perspectives for future research.

2 Communication of Mobile Agents in a Network

Fast communication of knowledge between mobile agents along a network can be done : directly, by messages exchange, when agent are physically close enough to communicate; or indirectly through the environment (generally the nodes are used to store / read information).

We think that a node should be more pro-active and should have the opportunity to choose to communicate some information to some agents chosen according to its knowledge.

So we propose a third method in which some non mobile agents are located at the nodes and communicate with mobile agents and that are close enough to receive messages.

Moreover, we can easily argue that to give the possibility to a node to interact with one another could give efficiency to the propagation of an information along a network.

Multilevel Architecture. In order to allow the three kinds of communication (‘*mobile-agent-to-near-mobile-agent*’, ‘*mobile-agent-to-near-node*’ and ‘*node-to-near-node*’), and to allow communication between distant mobile-agent and distant nodes, we use a multi-level architecture, inspired from holonic principles. We have already used this kind of architecture for the simulation of a flexible assembly cell, in order to correct myopic behaviour of mobile and autonomous shuttle [1].

At the bottom of the system (level 0), we have the mobile agents. At he level 1, there are the node agents, that can schedule, manage conflicts between ‘mobile agents’ that have to cross them. These agents (mobiles and nodes) and the environment (the network) are included in the ‘network agent’ that represents the multiagent system.

Each agent can interact with all agents that are in its ‘vision field’.

The Figure 1 presents an extract of a screenshot of a traffic simulator that we use to test knowledge communication between cars; ‘mobile agents’ are represented by ‘vehicle agents’, ‘node agents’ are represented by ‘crossroad managers’.

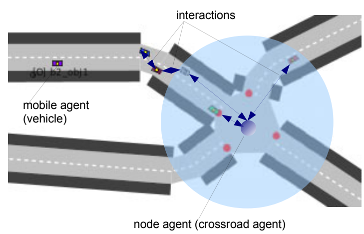

Fig. 1 Example of application in road traffic. Mobile vehicles agents interact between them and with the crossroad (node) agent if they are close enough.

Elements of Knowledge Model. We give here just some elements of how the mobile agents manage the exchanged knowledge. A knowledge is a partial view of the environment or of the other agents, namely for a given object o of the environment (the traffic network for example); it is (generally) an incomplete copy of it, so a representation of o with missing attributes and methods.
If $\mathscr{O}$ is the set of objects of the environment, an object o is defined by:

$$o = (id, attributes_o = \{a_1^o, \ldots, a_n^o\}_{n>0}, functions_o = \{f_1^o, \ldots, f_m^o\}_{m\geq 0}) \qquad (1)$$

with $f_j^o : \mathscr{O} \mapsto \mathscr{O}$.

If o'_a is a partial representation of o for the agent a, the cardinalities of the set of attributes and functions of o' are lesser than those of the object o:

$\mid attributes_{o'} \mid \leq \mid attributes_o \mid$ and $\mid functions_{o'} \mid \leq \mid functions_o \mid$

We define a knowledge κ_o^a (cf. def. 2) on an object o for an agent a by: o'_a, a partial view of o from a; $date_{\kappa_o}$, the date when the knowledge has been created or updated (by a or by another agent if the knowledge has been received); $builderAgent_{\kappa_o}$, the ‘builder’ of the knowledge (name of the agent that has created/updated the knowledge from its perception); $senderAgent_{\kappa_o}$, the ‘sender’ of the knowledge (name of the agent that could have sent the knowledge to a); $receiverAgents_{\kappa_o^a}$, the list of agents to which the knowledge has been sent by a; $shareable_{\kappa_o^a}$, the fact that the knowledge is shareable or not by a.

$$\kappa_o^a = \begin{pmatrix} o'_a, date_{\kappa_o}, builderAgent_{\kappa_o}, \\ senderAgent_{\kappa_o}, receiverAgents_{\kappa_o^a}, shareable_{\kappa_o^a} \end{pmatrix} \quad (2)$$

The notions of $date_{\kappa_o}$ and of $receiverAgents_{\kappa_o^a}$ allow to decrease the number of communications. Indeed, when the agent a perceives, from its perception function an image of a new object o, it creates the knowledge κ_o^a with the date of creation.

When a receives from another agent b a knowledge on o, a replaces its knowledge only if the $dateCreation_{\kappa_o}$ in κ_o^b is newer than the one in κ_o^a.

If b is close enough to send its knowledge to a, reciprocally a can send its knowledge. So when two agents meet, at the end of the communications they own the last representation known of the object o. When an agent (a) sends some knowledge on o to another (b), it adds the agent to its set of receivers ($receiverAgents_{\kappa_o^a} \leftarrow receiverAgents_{\kappa_o^a} \cup \{b\}$. Each time an agent receives a new version of an information, its set of agents to which this information has been sent is emptied ($receiverAgents_{\kappa_o^a} \leftarrow \{\}$). This allows to decrease the number of communications; for instance, when a and b meet many times (for example, if they follow the same path), the communication about a given knowledge is done only once.

3 Simulation

We illustrate our proposal on a traffic road simulator that we have developed in the context of a project (Plaiimob : a simulating Plateform dedicated to Mobility services) of CISIT (for International Campus on Security and Inter modality in Transports).

The aim of this project is to allow vehicle-to-vehicle (V2V) communication to allows drivers to automatically exchange data about their environment (incident/traffic jam on a road, information about off-street parking ...) [8].

We developed a traffic road simulator with the Jade Platform[1]:

- The *environment* is a traffic network in the OpenStreetMap format[2] (OSM). This allows us to use true traffic networks, or to define our own maps in order to test particular situations.
- The classes of *agents* are: the PersonnageAI agent class (it allows to simulate the behaviour of a driver), the CrossRoadManager agent class (it allows to manage the priorities at a crossroad according to the road signs), the ObserverAgent class (that allows to draw statistics from the simulating exercise).
- The *roles* played by the PersonnageAI are: RandomBehaviourRole: to represent a driver that moves randomly on the traffic; BusRole: to represent a driver that starts from a particular point and has to reach an objective, by linking some

[1] See the web site of the Jade platform : `http://jade.tilab.com/`

[2] See the web site of the OpenStreetMap project: `http://www.openstreetmap.org/`

bus stops, at earliest; EmergencyRole: to represent a driver of an ambulance, firetruck, for instance, that has an objective to reach as soon as possible.
- The *roles* played by the CrossRoad Managers are linked to the road signs or traffic lights associated with them. We defined some roles like: FifoCrossRoad (the first vehicle arrived in a queue of the crossroad 'receives' a green light); ClassicTrafficLightRole (emulation of classical traffic lights at each entry of the crossroad); AITrafficLightRole (based on the ClassicTrafficLightRole, it gives the priority to the street from which an emergency vehicle arrives (the associated traffic light goes on green light) ...).

Jade has been chosen because the aim of the project is to test our proposal on a real case, with agents embedded in the smart-phones of the drivers. So we aim at reusing some classes of the simulation to build agents and their roles.

Communication of Incident. In order to test the benefit to have a V2V communication, we present here a small case study that includes 2 bus lines (A and B), with one bus for the line A ($bus1a$) and 5 bus of the line B ($bus2a, \ldots, bus2e$); and five cars that travel randomly on the network ($randomCar1, \ldots, randomCar5$).

The Fig. 2 shows the map used in this context. The bus of the line A start from 'SA', and have to make loops between the bus stops 'BS-A1', and 'BS-A2'. For the busses of the line B, they start from 'SB', and have to make loops between the bus stops 'BS-B1', and 'BS-B2'. 'SR' is the starting point of the cars that travel randomly.

We used three different scenarios. In the first scenario, there is no incident, all the vehicles travel as they planed. In the second scenario, we add an incident on the road with incident risk before $bus1a$ reaches it, and vehicles do not communicate information. In the third scenario, busses and cars are able to exchange knowledge.

The incident causes a tailback on this road, and the speed limit is divided by 2 (so it becomes 35 km/h maximum).

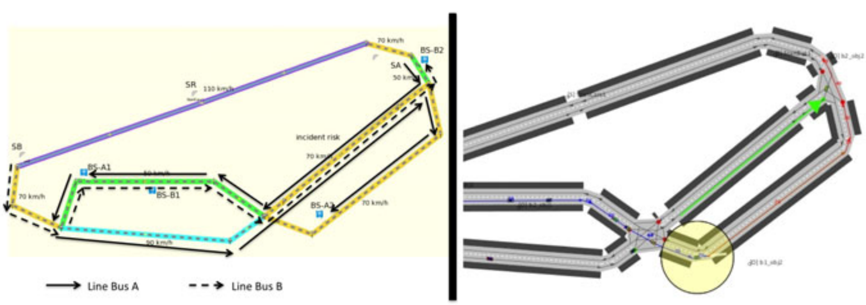

Fig. 2 On the right: map of Traffic Roads used in the case study. The busses have to stop on the right side of the road. One road is prone to incident risk. On the left: screencopy of the Traffic Roads Simulator. $bus2b$ is selected; the red arrow shows the roads where it plans to go. The disc around $bus2b$ represents its communicating zone.

Initially, each mobile agent knows the map of the traffic road, with the initial speed limit.

Each time a mobile agent reaches a road, it updates its knowledge if needed: if the perceived state of the road implies a modification of the speed limit (due to an incident, or traffic jam, for example), a knowledge is created and stored; if the perceived road exists in the knowledge, the knowledge is updated (the date is updated, the state of the road is modified if necessary).

If no incident is signaled, but if the agent took at least twice more time to exit the road, it records a tailback event in its knowledge.

Each 'PersonnageIA' with the role 'BusRole' is able to compute the shortest path to its next objective with the Dijkstra algorithm [3].

The best path, without incident, for A is to made a double loop (a '8') : from 'SA', the bus goes by the road with the 'incident risk'; then to 'BS-A1'; then it takes the road limited to $90km/h$; takes again the 'incident risk' road; to reach 'BS-A2' and repeats its round.

For the busses of the line B, the best path starts from 'SB'; goes to 'BS-A1'; takes the 'incident risk' road to reach 'BS-A2'. The busses take next the fast lane to repeat their round.

In scenario 3, we choose to put an incident at the beginning of the simulation; $bus1a$, which does not know this fact, travels on the route with incident. It informs all the agents that it meets during its move. $bus2b$ is the first informed by $bus1a$, it computes a new best path and chooses another road than the one taken by $bus2a$.

On Fig. 2-right, the green arrow shows the road took by $bus2a$, the red arrow shows the road took by $bus2b$ (and its followers) that met $bus1a$.

Without communication between vehicles, the lost time when an incident occurs on the 'incident risk' road is 7 time units on average. The communication of information, in the particular case study, results in a time-saving of about 4 time units.

Communication of Incident Repairing. In the simulator, when a incident is removed from a road, this one goes back to its initial speed limit. If all the busses have got the knowledge that an incident has occurred on the road, they will never takes this road again. They could get the information about the repairing only if a vehicle with a RandomBehaviourRole take this road and meet the busses.

Results from the simulation about this propagation of incident repairing is not really pertinent, because it depends on the random choice of path by the vehicles.

So if a simulation does not include 'random cars', once a path is declared 'with incident', and once the event has been propagated to all mobile agents, the path is neither reused. It is important to allow agents to be informed about a repairing; so it is important to allow at least one mobile agent to perceive this repairing. A mobile agent, having the information that an incident has occurred on a path, has to decide to take it, after *some* time.

We introduce a notion of degradation, in the definition of a knowledge κ_o^a. We define $degradation_{\kappa_o}$ as a coefficient of the reliability of the knowledge (if t is the elapsed time since the creation/update of the knowledge, the confidence that a has on it is: $trust_{\kappa_o} = 1 - t \times degradation_{\kappa_o}$). For a knowledge κ_o, if $trust_{\kappa_o} \leq 0$, then the agent does not take into account this knowledge to compute the best path to its next objective; and the knowledge is not sent to other agents.

If $degradation_{\kappa_o} = 0$, then the knowledge κ_o is perennial; if a $degradation_{\kappa_o} = 0.1$, then the knowledge κ_o exists for 10 tu.

The degradation coefficient for a knowledge is strongly dependent of the case study. Here, due to the light complexity of the traffic network used, as a bus needs 2.2 tu to make a loop, we fix the degradation coefficient of a knowledge about an incident at $1/3$ tu. This allows a bus to try to pass by the 'incident road' after 3 loops; if the incident is always on the road, the date of the knowledge is updated, and the new knowledge is communicated to each vehicle met by the bus.

One perspective could be to automatically define the degradation coefficient from observation during the simulation.

Propagation of Knowledge. In order to improve the diffusion of knowledge, we propose to use agents linked to the nodes of the network (here, these agents are the CrossRoad Managers).

When a mobile agent decides to communicate its knowledge, it includes the local node agents in its potential recipients. All nodes are not necessary equipped with node agents, but in the case study, a CrossRoad agent is located to each crossroad. When a vehicle detects a problem on the road it reaches, the vehicle informs all agents inside its perimeter of communication. The crossroads at the beginning and the end of the road receive the information. When another vehicle approaches the road, the crossroad where the vehicle arrives communicates the knowledge on the road incident/repairing. The vehicle is able to recompute its best path, and to choose another road.

We define so a fourth scenario with crossroad manager agents able to communicate knowledge. The results of the simulation is the same as the scenario 3, except that it is not needed to have a line A bus to inform line B busses about the incident on the 'incident risk' road.

4 Conclusion

In order to allow the propagation of knowledge between mobile agents, with a minimum of exchanged messages, we propose an architecture, a model of knowledge and a model of communication.

In the first results presented in this paper, we make the assumption that all the agents are cooperative, and no deficient; that is to say that they cannot send wrong knowledge, voluntary, or not (if a captor has a dysfunction).

The trust on a knowledge depends only on the date from which the knowledge has been updated or created. We plan to introduce the notion of trust that depends of the sender; for example, if an agent t sends a knowledge that is not coherent with the perception of other agents, these latters can decide to put in quarantine the faulty agent, like in [10] for example.

We plan also to enhance the information sent by the crossroad agents: when a car informs a 'crossroad agent' about its desire to take a particular road, the crossroad manager sends the number of agents that have already taken this road in a near past. So the mobile agent can choose to pursuit its initial plan, or to recompute a new best path.

Acknowledgments. This research was financed by the International Campus on Safety and Inter-modality in Transportation, the Nord/Pas-de-Calais Region, the European Community, the French Regional Delegation for Research and Technology, and the French National Centre for Scientific Research. We are grateful for the support of these institutions. The authors thanks particularly Remy Tylski, who implemented interesting parts of the simulator.

References

1. Adam, E., Zambrano, G., Pach, C., Berger, T., Trentesaux, D.: Myopic Behaviour in Holonic Multiagent Systems for Distributed Control of FMS. In: Corchado, J.M., Pérez, J.B., Hallenborg, K., Golinska, P., Corchuelo, R. (eds.) Trends in PAAMS. AISC, vol. 90, pp. 91–98. Springer, Heidelberg (2011)
2. Bazzan, A.L.: A distributed approach for coordination of traffic signal agents. Autonomous Agents and Multi-Agent Systems 10, 131–164 (2005)
3. Dijkstra, E.W.: A note on two problems in connexion with graphs. Numerische Mathematik 1, 269–271 (1959)
4. Dresner, K., Stone, P.: Mitigating catastrophic failure at intersections of autonomous vehicles. In: AAMAS Workshop on Agents in Traffic and Transportation, Estoril, Portugal, pp. 78–85 (2008)
5. Ferber, J.: Multi-agent systems - an introduction to distributed artificial intelligence. Addison-Wesley-Longman (1999)
6. Jennings, N.R., Sycara, K., Wooldridge, M.: A roadmap of agent research and development. Int. Journal of Autonomous Agents and Multi-Agent Systems 1(1), 7–38 (1998)
7. Mandiau, R., Champion, A., Auberlet, J.-M., Espié, S., Kolski, C.: Behaviour based on decision matrices for a coordination between agents in a urban traffic simulation. Appl. Intell. 28(2), 121–138 (2008)
8. Popovici, D., Desertot, M., Lecomte, S., Peon, N.: Context-aware transportation services (cats) framework for mobile environments. IJNGC 2(1) (2011)
9. Ruskin, H.J., Wang, R.: Modelling Traffic Flow at an Urban Unsignalised Intersection. In: Sloot, P.M.A., Tan, C.J.K., Dongarra, J., Hoekstra, A.G. (eds.) ICCS-ComputSci 2002, Part I. LNCS, vol. 2329, pp. 381–390. Springer, Heidelberg (2002)
10. Vercouter, L., Jamont, J.-P.: Lightweight trusted routing for wireless sensor networks. In: Demazeau, Y., Pechoucek, M., Corchado, J.M., Pérez, J.B. (eds.) PAAMS. AISC, vol. 88, pp. 87–96. Springer, Heidelberg (2011)

Nervousness in Dynamic Self-organized Holonic Multi-agent Systems

José Barbosa, Paulo Leitão, Emmanuel Adam, and Damien Trentesaux

Abstract. New production control paradigms, such as holonic and multi-agent systems, allow the development of more flexible and adaptive factories. In these distributed approaches, autonomous entities possess a partial view of the environment, being the decisions taken from the cooperation among them. The introduction of self-organization mechanisms to enhance the system adaptation may cause the system instability when trying to constantly adapt their behaviours, which can drive the system to fall into a chaotic behaviour. This paper proposes a nervousness control mechanism based on the classical Proportional, Integral and Derivative feedback loop controllers to support the system self-organization. The validation of the proposed model is made through the simulation of a flexible manufacturing system.

José Barbosa · Paulo Leitão
Polytechnic Institute of Bragança, Campus Sta Apolnia,
Apartado 1134, 5301-857 Bragança, Portugal
e-mail: {jbarbosa,pleitao}@ipb.pt

José Barbosa · Emmanuel Adam · Damien Trentesaux
Univ. Lille Nord de France, F-59000 Lille, France
e-mail: {emmanuel.adam,damien.trentesaux}@univ-valenciennes.fr

José Barbosa · Damien Trentesaux
UVHC, TEMPO research center, F-59313 Valenciennes, France

Paulo Leitão
LIACC - Artificial Intelligence and Computer Science Laboratory, R. Campo Alegre 102, 4169-007 Porto, Portugal

Emmanuel Adam
UVHC, LAMIH, F-59313 Valenciennes, France

Emmanuel Adam
CNRS, FRE 3304, F-59313 Valenciennes, France

J.B. Pérez et al. (Eds.): Highlights on PAAMS, AISC 156, pp. 9–17.
springerlink.com

1 Introduction

The need to develop more flexible and adaptive factories that better addresses the current requirements imposed to manufacturing companies, such as flexibility, reconfiguration and responsiveness, is an important issue in the Factory of the Future (FoF) program and implies the adoption of new production control structures. Holonic manufacturing and multi-agent systems [4] are examples of suitable approaches that overcomes the typical problems exhibited by traditional centralized control structures, e.g. the rigid, monolithic structures. In fact, they offer an alternative way to design control systems based on a set of distributed, autonomous entities that may cooperate to reach the systems' goals. These distributed entities, the holons, have a local perspective of their world surrounding and may suffer from information deficiency and information quality, meaning that any decision is based on incomplete and even, at some point, inaccurate/outdated information. In these holonic systems, holons take decisions in a myopic manner relying only of the existing local information.

Self-organization properties [1] embedded in holonic manufacturing systems, can contribute to improve the system performance. In the same way, in such dynamic and continuous self-adaptive systems, some undesired instability, due to the presence of nervousness, may appear and should be taken into consideration. Nervousness can be characterized by or showing emotional tension, restlessness and agitation, i.e. as the frequency that a holon changes its intentions. As expected, a repeated changing of plans can originate undesired behaviour, leading to a loss of the system performance, and in extreme cases, into pitfalls or chaotic behaviour. Since the objective is to push these systems to their limits but maintaining them under control, the study of the system nervousness is crucial to design dynamic self-organized holonic and multi-agent systems, and particularly, how to control the nervousness of each individual holon.

This paper discusses the control of nervousness in dynamic self-organized holonic structures, namely overcoming the reasons for the occurrence of instability in distributed adaptive systems and identifying the benefits and liabilities associated to nervousness. A nervousness control mechanism is proposed based on the classical Proportional, Integral and Derivative (PID) feedback loop controllers. Aiming to validate the proposed approach, an experimental manufacturing cell was simulated using a self-organized agent-based system embedded with the proposed nervousness mechanisms. The rest of the paper is organized as following: the next section explains the importance of having reactive holons balancing the local and global aspects of the self-organized distributed system. Section 3 depicts the PID based proposed model for the nervousness control to be embedded in each holon, and section 4 describes the experimental case study. Finally, section 5 rounds up the paper with the conclusions and future work.

2 Nervousness and Equilibrium in Dynamic Systems

The new manufacturing control paradigms have foundations in decentralized control over distributed entities, i.e. the global control is distributed by the entities in the system. The decentralized nature of these systems allows the achievement of responsiveness to external perturbations since they can locally react to the disturbance in a faster way. This situation turns on the need of the system to display self-organization properties allowing an automatic arrangement of the system entities to allow the system to fulfill its goals.

Latest concept developments of self-organization state that self-organizing systems include not only adaptation and learning mechanisms but also stable structures. Additionally, self-organization can also be achieved by the global cooperation in dynamical systems by means of emergence properties which consist of the multiple lower-level interactions, in one hand by entities-entities interactions and on the other by entities-environment interactions. A deep study of this issue can be found in [3].

In such dynamic self-organized systems, some instability may occur, resulting of the nervousness of the individual entities. For this purpose, the system nervousness should be balanced trying to push the system into its limits but maintaining it under control, allowing the system to dynamically evolve safely into different structures maintaining high performance levels. This idea is in line with the application of theory of chaos as seen in [6].

In this work, nervousness is characterized by the frequency that an entity changes intentions during its life-cycle. Naturally, an entity is nervous if it is constantly changing its intentions and in opposite an entity is calm when does not change intentions. The pertinent question is how to know that a new solution can bring better results than the previously one and in the case of distributed systems if it will negatively affect the system performance. Also vital is to translate this phenomenon into the system design allowing to push it to its limits keeping it under control.

Parunak et al. use two mechanisms to calm hyperactive agents [8]. The first one uses a stigmergy like mechanism used by societies of insects, allowing to merge multiple local data in order to take decisions. The second mechanism delineates a way to separate exploration and exploitation times relating them to the imposed deadline time.

Having in mind the same objective, Hadeli et al. state that, in case an agent wants to change the initial ideas, it is necessary to make a quality measurement of the new solution and only if the new solution is significantly better than the previous one the agent can change its intentions [5]. Additionally, the frequency of changing the initial intentions is limited, restringing the continuous willing to change. Another example can be found in [9] where the inspiration from magnetic fields helps products to choose the most adequate route. In this mechanism, as resources are being occupied, the attraction power emitted (designated by potential field) is reduced allowing products to route in a decisive manner.

The referred approaches allow to control the nervousness phenomena; however the goal of this paper is to go further and introduce a nervousness mechanism that, in a natural manner, is associated to the control mechanism and simultaneously supports the dynamic self-organization.

3 A PID Based Approach for the Nervousness Control

In the classical feedback control theory, one of the most and effective mechanism used to control discrete or continuous systems is the PID controller [7], namely its discrete version which allows to be implemented into more sophisticated processing units. This mechanism, adjusting some key parameters, allows a quick reaction to the perturbation combined with the elimination of the error in a steady state. In practice, when different functioning conditions are needed, e.g., a different temperature in a room, the PID controller adjusts in a quick and effective way, by setting the variables to the new set-point. Having in mind these principals, the inspiration of the PID control was used to design a nervousness control mechanism for self-organized systems, as illustrated in Fig. 1.

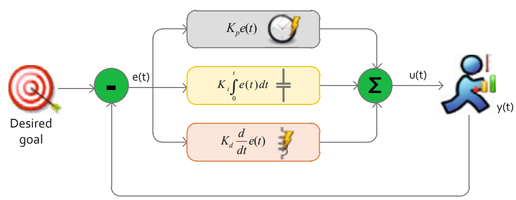

Fig. 1 PID controller block diagram for the nervousness control

The stability and equilibrium analysis can be performed in analogy to the transient response or natural response behaviour. The response can be classified as one of four types, describing the system output in relation to the steady-state value (in this case, the adaptation response): *Overdamped* response that does not oscillate about the steady-state value but is slow to reach it, i.e., exhibits a slow reaction to the perturbation; *Critically damped response* that reaches the steady-state value the fastest without being underdamped - in an ideal case, there should be no oscillation about the steady state value; *Underdamped response* that oscillates within a decaying envelope, exhibiting more oscillations and longer to reach steady-state being as more underdamped is the system - it presents a fast adaptation response but is nervous and can lose the stability; *Unstable response*, where the system is not able to respond to the set point and gets uncontrolled.

Since the objective is to avoid the system instability, the last type of response can make the system fall into a chaotic behaviour and should be avoided. The achievement of a fast adaptation response maintaining the system stable and predictable is a balance between adjusting the damping parameter of the system/holons. Note that if the system has overdamping, the system response may be inadequate, i.e., too slow, due to the time needed for adaptation.

The question is how to handle the local behaviours to reach more adaptive systems with less nervousness. Using the analogy with the PID controllers found in the control theory, the idea is to adjust the coefficients K_p, K_i and K_d that define the local behaviour of the holons, as illustrated in the equation (1).

$$f(t) = K_p e(t) + K_i \int_O^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \tag{1}$$

In the proposed approach, the meaning of each component is: K_p, i.e., the proportional part, is related to the time to react to the perturbation (meaning the adaptation response); K_i, i.e., the integral part, is related to the error in steady-state state (meaning reaching the optimal state); K_d, i.e., the derivative part, is related to the speed of the stabilization eliminating the error (meaning the responsiveness of achieving the optimal state).

Adjusting each parameter of the controller will culminate in the desired performance curve for a given system. The consequences and cautions that must be taken when adjusting the parameters can pass by the decrease of rise time, increase of overshoot, increase of the steady state error and a degradation of the system stability. Individualizing and extrapolating into holonic multi-agent systems, the increase of K_p can lead to the increase of the overshoot and the degradation of the stability, that can be seen as a temporary difference to the desired behaviour (see Fig. 2_{left}). The K_i parameter drive the system to have higher overshoot and settling time which is counterbalance by a significant decrease of the steady state error, leading the system to its goal with more accuracy (see Fig. 2_{middle}). Finally, the K_d parameter can influence more the system stability but improves the overshot, allowing the entity/system not to greatly drift from desired behaviour in initial time (see Fig. 2_{right}).

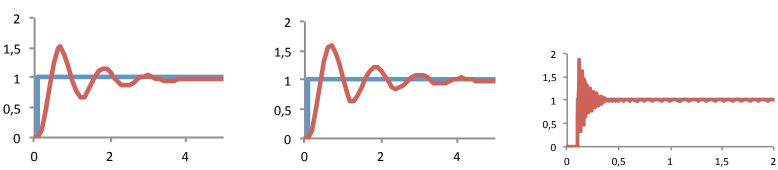

Fig. 2 Response curves adjusting K_p, K_i and K_d

The stability and the convergence to the global optimization can be affected if the individual entities self-learn, self-adapt and self-organize taking different directions. Nevertheless, self-learning capabilities embedded into the individual behaviours can gather valuable information that assists the auto-tuning of the PID control parameters on-the-fly (i.e. without the need to stop, re-program and re-start the system). Although the PID controller parameters can be adjusted in an offline mode, allowing an easier development, the controller has more potentialities if the parameters are adjusted in real time to face the changes of functioning conditions.

In distributed systems, an important question is how to guarantee the overall system equilibrium based on the control of the nervousness of the individual entities. Being the proposed model more suitable to be developed into individual entities, a global unifying solution pass by self-organization consideration such as found e.g., in societies of ants. Alternatively, rules could be deployed in which the eager that individual entities have to fulfill their goals decrease as those goals are being accomplished. The consequence would be that entities would decrease their nervousness levels alongside the goals fulfillment.

4 Case Study

The proposed model, described in the previous section, was used to solve the nervousness of a self-organized agent-based system applied to route pallets in a flexible manufacturing system. This section describes the case study and the experimental tests performed by using the Netlogo framework[1].

System Description. The flexible manufacturing system, illustrated in Fig. 3, is composed by two independent cells each one having two conveying levels joined by lifters. The first level is comprised by 9 interconnected conveyors while the lower level has 2 conveyors. In each cell, the left lifter is able to transport the product in an upper way and the right lifter in the opposite direction. The 2 cells are joined at both levels i.e., in the upper level the pallets can forward from the first cell to the second cell and in the lower level the pallets can leave the second cell re-entering the first cell. The global entrance of the system is made in the first cell through the upper level in the left lifter, being the output made on the right lifter of the second cell.

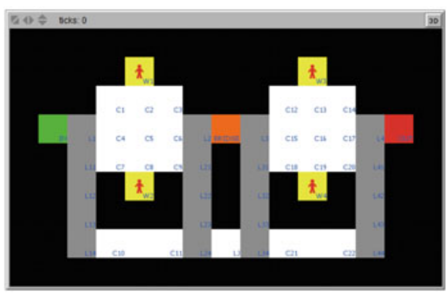

Fig. 3 Global view of the system. Workstations: $WS_1 = [S_1, S_2]$; $WS_2 = [S_1]$; $WS_3 = [S_2, S_3]$; $WS_4 = [S_2]$. List of products: $A = [S_1 S_2 OUT]$; $B = [S_3 OUT]$

The manufacturing system comprises 4 workstations, associated with conveyors C_2 (WS_1) and C_8 (WS_2), for the first cell, and C_{13} (WS_3) and C_{19} (WS_4) for the second cell. Each workstation performs a set of services, as described in Fig. 3.

[1] see http://ccl.northwestern.edu/netlogo/